University of Alberta

Library Release Form

Name of Author: Yan Xin

Title of Thesis: High-Order Spectral-Null Multimode Codes

Degree: Doctor of Philosophy

Year this Degree Granted: 2002

Permission is hereby granted to the University of Alberta Library to reproduce single copies of this thesis and to lend or sell such copies for private, scholarly or scientific research purposes only.

The author reserves all other publication and other rights in association with the copyright in the thesis, and except as herein before provided, neither the thesis nor any substantial portion thereof may be printed or otherwise reproduced in any material form whatever without the author's prior written permission.

University of Alberta

High-Order Spectral-Null Multimode Codes

by

Yan Xin ©

A thesis submitted to the Faculty of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy

Department of Electrical and Computer Engineering

Edmonton, Alberta

Spring 2002

University of Alberta

Faculty of Graduate Studies and Research

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research for acceptance, a thesis entitled HIGH-ORDER SPECTRAL-NULL MULTIMODE CODES submitted by YAN XIN in partial fulfillment of the requirements for the degree of DOCTOR OF PHILOSOPHY.

To my mother SUN Xiuying, my father XIN Dianchen, and
my wife ZHAI Fengqin

ABSTRACT

Constrained codes are employed in digital magnetic and optical storage systems to ensure that the recorded signals match the characteristics of the recoding channels to improve system performance. This thesis introduces a new class of constrained codes, high-order spectral-null multimode codes.

First, characteristics of high-order spectral-null codes are summarized, and further characteristics of these codes are investigated. A new method for enumerating codewords with specified values of word-end disparity and word-end disparity sum is introduced. These codewords are of interest for the design of second-order spectral-null codes. Enumeration rules are given, and two new efficient algorithms for evaluating the cardinality of codewords based on this enumeration method are derived. Calculation results demonstrate the advantages of the new enumeration algorithms.

Evaluation of power spectra of fixed-length and variable-length codes is then reviewed. This thesis demonstrates that sum variance, a performance metric for first-order spectral-null codes, cannot be generalized to be a valid metric for higher-order spectral-null codes and proposes a new performance metric, low-frequency spectrum-weight, for evaluation of an arbitrary high-order spectral-null code. It is shown that the asymptotic low-frequency spectral components of high-order spectral-null codes are exclusively determined by the order of spectral null and the low-frequency spectrum-weight. Low-frequency spectrum-weight of state-dependent and state-independent codes, and fixed-length and variable-length high-order spectral-null codes is evaluated. Several closed-from expressions are derived for the low-frequency spectrum-weight of specific classes of codes.

A state-dependent encoding method is proposed to construct rate-efficient high-order spectral-null bimode and multimode codes. The guided scrambling multimode coding technique is summarized, and a new serial encoder structure is proposed. Necessary and sufficient

conditions for the generation of complementary quotients in each guided scrambling selection set are derived. Scrambling polynomials suitable for implementation of high-order spectral-null multimode codes are obtained through further computer search.

The analysis of low-frequency spectrum-weight of the equivalent state-independent variable-length encoder provides insight into the construction of good fixed-length high-order spectral-null codes as well as good first-order spectral-null codes with a large rejection of spectral components at low frequencies. Spectral results of various high-order spectral-null multimode codes introduced in this thesis demonstrate that the new codes can yield superior spectral performance and higher rate codes than other high-order spectral-null codes developed to date.

ACKNOWLEDGEMENTS

I gratefully acknowledge the following people and institutes:

- ▲ Dr. Ivan J. Fair, for providing me an opportunity and offering financial supports to do this work, supervising the research, sharing his technical thoughts, and supplying a free space for research;

- ▲ the members of the examining committee, for taking the time to review this work;

- ▲ TR*Labs*, the University of Alberta, and Dalhousie University, for financial assistance;

- ▲ staff and students at TR*Labs*, for sharing the enjoyable time with me and affording support, in particular: Linda Richens, Rhoda Hayes, Luke Chong, Matthieu Clouqueur, David Mazzarese, Robert Novak, Ge Li, and Dr. Kwok-Sing Cheng;

- ▲ Prof. Zheng Li, who encouraged me to pursue my doctorate;

- ▲ my parents, for their deep love, understanding, encouragement, and support throughout my education;

and, especially my wife, Ms. Fengqin Zhai, for her love, encouragement, and support as well as for her technical discussion, comments, and verification.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

$\mathcal{A}(N,K)$ the set of Kth-order spectral-null words with length N

$|\mathcal{A}(N,K)|$ cardinality of the set of Kth-order spectral-null words with length N

$\mathcal{I}$ finite alphabet

$\{\boldsymbol{x}\}$ code symbol sequence

$\{\boldsymbol{y}\}$ source symbol sequence

$\{s_n\}$ state sequence

$\mathbb{X}$ the set of output symbols

$\mathbb{Y}$ the set of input symbols

$\mathbb{X}^\lambda$ the set of output length-λ words

$\mathbb{Y}^\mu$ the set of input length-μ words

g state transition function

h output function

σ states in finite-state sequential machine

Mbits/s 10^6 bits per second

Gbits/s 10^9 bits per second

(d,k) d represents the minimum length of runs of consecutive logical "zeros" between two consecutive logical "ones"; k represents the maximum length of runs of consecutive logical "zeros" between two consecutive logical "ones"

$(0,G/I)$ G denotes the global constraint and I denotes constraint in the even index and odd index

T_b duration of channel symbols

M/N code rate for block codes with binary source symbols and coded symbols

MB/NT code rate for block codes with binary source symbols and ternary coded symbols

$N^*(K)$ the minimum word length of a Kth-order spectral-null word

C rate capacity

$\lambda_{\max}$ largest eigenvalue of the state transition connection matrix

$\mathrm{RDS}^{(K)}$ Kth-order running digital sum

$H_x(\omega)$ power spectrum density of a symbol sequence

$H_{\mathrm{RDS}^{(K)}}(\omega)$ power spectrum density of an $\mathrm{RDS}^{(K)}$ sequence

$s(t)$	pulse shape signal
$X(t)$	pulse amplitude modulation signal
$E\{\cdot\}$	statistical average

LIST OF ABBREVIATIONS

AMI	alternate mark inversion
AWGN:	additive white Gaussian noise
CD	compact disk
dc	direct current
DAT	digital audio tape
DVD	digital versatile disc
FSSM	finite-state sequential machine
FSTD	finite-state transition diagram
g.c.d	greatest common divisor
GS	guided scrambling
HOSN	high-order spectral-null
K-OZD	Kth-order zero-disparity
ISI	intersymbol interference
i.i.d.	independent identically distributed
LDPC	low-density parity-check
LFSW	low-frequency spectrum-weight
ML	maximum likelihood
PAM	pulse amplitude modulation
PR	partial response
PSD	power spectrum density
RDS	running digital sum
RLL	runlength limited
RS	Reed-Solomon
SNR	signal-to-noise ratio
WD	word-end disparity
WDS	word-end disparity sum

Chapter 1
INTRODUCTION

Modulation codes, also referred to as constrained codes, are employed in digital magnetic and optical recoding systems to ensure that the recorded signals match the characteristics of the physical channels so as to improve the performance of the systems. This thesis introduces a new class of modulation codes called *high-order spectral-null* (HOSN) *multimode codes*, and investigates the characterization, implementation, and performance of the new codes. In this thesis, the proposed HOSN multimode codes generated through the *guided scrambling* (GS) multimode coding technique demonstrate both high rate-efficiency and excellent spectrum performance. The original research contributions of this thesis include:

- a systematic construction method for generating rate-efficient high-order spectral-null codes;
- fast algorithms for enumerating the number of codewords for second-order spectral-null constrained channels;
- improved bounds on the minimum word length and redundancy of HOSN words;
- a general performance metric for analysis and design of codes with an arbitrarily high-order spectral null at zero frequency, as well as alternative methods to evaluate the metric values of various specific cases;
- scrambling polynomials in GS suitable for HOSN multimode codes;
- codeword selection criteria for HOSN multimode codes;
- evaluation of power spectral density of HOSN multimode codes.

In the remainder of this chapter, the primary channel constraints and corresponding coding strategies in digital recording systems are briefly described. The evaluation of coding schemes is then considered, and an overview of this thesis is given.

1.1 Constrained Coding in Digital Recoding Systems

Storage and retrieval in a digital recording system can be considered as transmission and recovery of digital information in a digital communication system over a special channel [1, ch.1]. To paraphrase [2]: "Communication links transmit information from here to there. Recording systems transmit information from now to then."

Since the early 1990s, storage capacity and data transfer rate in digital recoding systems have grown rapidly. Commercial disk drives currently store information at densities of up to 18 Gbits

per square inch and stored data can be transferred at a rate of up to 700 Mbits/s [3]. These storage systems require very high reliability. Typical bit error rate requirements are on the order of 10^{-14}, and the probability of miscorrection is on the order of 10^{-21} [3]. Signal processing and coding play a key role in accurately retrieving data in advanced digital storage systems.

Figure 1.1 shows a block diagram of the coding steps involved in digital recording systems. In general, channel coding includes concatenation of error-correction coding and modulation coding, where an error-correction code forms the outer code and a modulation code is the inner code [4]. The aim of coding and digital signal processing in storage systems is to ensure reliable transfer of as much information as possible through the storage media. A digital recording channel is modeled as a linear intersymbol-interference (ISI) binary-input channel with additive white Gaussion noise (AWGN) [4]. The channel signaling representatives are the positive and negative magnetization in magnetic recording systems, or pits and lands in optical recording systems.
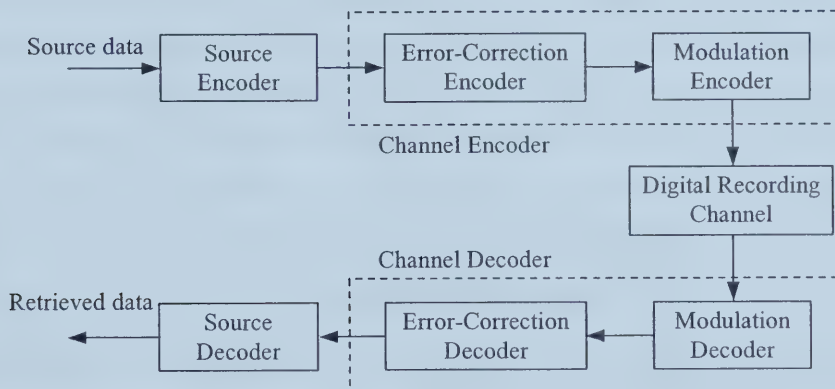
Figure 1.1   Block diagram of coding steps in digital recording systems.

The error-correction codes most often employed in current commercial storage systems are the well-known powerful Reed-Solomon codes [1], [5] which correct errors after detection and decoding. Recently introduced *turbo codes* [6] and *low-density parity-check* (LDPC) *codes* [7] have also attracted interest in storage systems [3].

In addition to error-correction coding, digital storage systems also employ constrained coding, which is addressed in this thesis, for improving timing recovery, reducing intersymbol interference between recorded symbols, and controlling the spectral content at specified frequencies, among other characteristics. In digital storage systems, channel constraints are a result of channel bandwidth restrictions, the timing recovery method, the gain control method,

the detection technique, and the electronic implementation approach [8]; these channel constraints can be described either in the time domain or in the frequency domain. Various coding techniques have been derived to ensure that the channel input sequences are compatible with these channel constraints. Several of these techniques are discussed below.

A. Runlength-limited constrained codes

In commercial digital magnetic and optical recording systems at low to moderate densities, an analog detection method, *peak detection*, is used [4]. To reduce the effects of ISI and clock phase drifting in peak detection, a class of constraints called *runlength-limited* (RLL) or (d, k) constraints are considered [9]. A (d, k)-encoded binary sequence satisfies the constraint that runs of consecutive logical "zeros" have length at least d and at most k between two logical "ones", where d and k are nonnegative integers and $d \leq k$. Owing to subsequent differential encoding, here the logical "one" and "zero" are indicative of a transition and a non-transition of binary symbols in the channel sequence, respectively. The minimum runlength parameter d limits the highest transition frequency of the sequence, in order to mitigate the effect of ISI caused by the bandwidth-limited channel and realize simple detection of the positions of the pulse peaks. The maximum runlength parameter k ensures that there are enough transitions in the sequence for clock recovery. For a sequence with no RLL constraints, d is zero and k is infinite. A (d, k) code realizes a mapping between an unconstrained $(0, \infty)$ sequence and a (d, k) sequence. An example of an RLL constrained sequence satisfying $(d, k) = (1, 7)$ is:

$$\cdots 01000100000001001010000 \cdots .$$

Efficient (d, k) codes are designed to increase the density of information stored along paths (tracks) in recording media [1, ch.4-8, ch.14], [10].

B. PRML constrained codes

In contrast to the peak detection, current high-density recording systems employ a digital detection technique denoted PRML [4], [11] which is based on *partial response* (PR) [12] equalization with *maximum-likelihood* (ML) sequence detection [13]. It has been shown that partial-response signaling can result in high storage densities without excessive loss of signal-to-noise ratio (SNR) [14]. This gain in storage densities is obtained only with the penalty of the increased complexity of the Viterbi detector [13], [15].

Recoding systems with different detection methods may have different constraints. For PR4-based PRML detection [11], the minimum runlength is unconstrained, however there is a G

constraint which plays the same role as the k constraint in a (d, k) sequence, limiting the maximum number of consecutive logical "zeros" between two consecutive logical "ones". In addition, there is another constraint, denoted the I constraint, which is "a separate 'interleaved' k constraint on the even index and odd index subsequences" [8]. This class of constraints for PRML channels is represented as a $(0, G/I)$ constraint. An example of a $(0, G/I)$ constrained sequence satisfying a $(0, 3/4)$ constraint is:

$$\cdots 10001100101000110101001 \cdots .$$

General coding schemes for generation of $(0, G/I)$ constrained sequences have been introduced in [8].

C. *Spectral-null codes*

For digital storage systems, in addition to the (d, k) RLL constraints and the $(0, G/I)$ PRML constraints, both of which are represented by the parameters in the time domain, there is another family of constraints which require that the binary channel input sequences have spectral nulls at predetermined frequencies.[1] That is, the *power spectral density* (PSD) function (power spectrum in brief) of the sequence has zero value at the specified frequencies. Spectral-null codes ensure that the encoded sequences satisfy the required spectral-null constraints.

Coded sequences with zero spectral content at zero frequency are of most interest in digital recording [4]. Coding techniques that generate these sequences are often called *dc-free* codes. Dc-free codes were originally developed in digital communication systems for cable and optical fiber baseband channels to reduce the effects of low-frequency cut-off due to coupling components, isolating transformers, etc. [16], and have also found widespread applications in digital magnetic and optical recording systems [1, ch.9-11, ch.13-14].

In current magnetic systems, binary digital information is stored by saturation recording [4] where magnetic particles on a magnetic disk or tape are magnetized in one of two directions through a write head. Dc-free codes are employed in these systems to prevent signal distortion caused by the use of transformers in write electronics.

In both magnetic and optical disc systems, information is recorded along tracks on discs. Simple servo systems for following and focusing on the tracks are realized by inserting tracking information in the form of low-frequency pilot tracking tones into the media, and by extracting the information while reading the data [1, ch.12], [17]. Dc-free codes are used in these systems

---

[1] Typically the specified frequency is at zero frequency or at the Nyquist frequency (one-half the symbol frequency of the channel input sequence). It is also possible for a channel-input sequence to have no spectral content at more than one frequency simultaneously.

4

to reduce the interference between data and servo signals. In addition, the use of spectral-null codes makes it possible to filter low-frequency noise caused by impairments such as fingerprints and dust on the disk surface, without eliminating important components of the signals [10].

Because the power spectral density of a real-valued coded sequence is an even function of frequency, all its odd derivatives are zero at dc. When there is a null at dc but the second and higher even derivatives of the power spectrum are non-zero at dc, the code is said to have a first-order spectral-null at dc. As a result, this subclass of dc-free codes is denoted as the class of *first-order spectral-null* codes or dc^1-*free* codes. First-order spectral-null codes are successfully used in most of the current commercial digital storage systems such as the compact disk (CD), MiniDisc, digital audio tape (DAT), and digital versatile disc (DVD) [1]. When the power spectrum and one or more of its even derivatives are zero, the code has a high-order spectral null at dc.

In digital storage systems, generally more than one constraint exists, for instance, dc-free and runlength limited constraints. Several coding schemes designed for these kinds of constraints in practical optical disk recording have been surveyed in [10], and other coding methods have been proposed in [18]-[20] to target higher code rate as well as larger low-frequency suppression.

To implement interference-free tracking and efficient low-frequency noise filtering, coded sequences need significant rejection of low-frequency components near dc [10]. Recently proposed dc^1-free multimode codes [21] have demonstrated advantages over other dc^1-free codes. Alternatively, owing to their superior performance in terms of rejection of spectral components at low frequencies [22] and distance-enhancing properties [23]-[25], high-order spectral-null codes have attracted considerable interest [1, ch.11], [22]-[31]. This thesis presents a new class of dc-free codes, called high-order spectral-null multimode codes, with both high rate-efficiency and large rejection of low-frequency components.

In the following, alternative considerations regarding the design of good dc-free codes are overviewed, based on which efficient high-order spectral-null multimode codes are constructed in this thesis.

1.2 Assessment of Coding Schemes

In the design of high-order spectral-null codes, it is desirable that a good coding scheme has the advantages described below.

• *High code rate*. Shannon [32] established the well-known formula for the capacity of a constrained noiseless channel. The capacity provides an upper bound for information transfer

rate in a coding scheme that translates unconstrained sequences into sequences with constraints. A closed-form expression for capacity of dc[1]-free codes was given in [33]; capacity of codes with a higher-order (up to the third order) spectral null at dc was calculated and listed in [25]. The rate of a HOSN code is desired to be as high as possible. In this thesis, rate efficiency is denoted as the ratio of the code rate to the capacity.

- *Good spectrum performance.* As described above, power spectra of the family of dc-free codes have spectral nulls at zero frequency. It is desired that a dc-free code results in large suppression of low-frequency components in order to satisfy the requirements of the storage systems.

- *Simple encoder and decoder.* Ease of implementation of the encoder and decoder results in fast encoding and decoding as well as a reduction in implementation cost. However, the simple encoder and decoder must result in sequences that satisfy the performance requirements.

- *Small error propagation.* In general, state-dependent encoding can realize higher code rates than state-independent encoding [22], [25]. However, it is preferred that a decoder be state-independent [4] because an error in the coded sequence may easily cause a state-dependent decoder to lose track of the encoder state sequence, resulting in significant error propagation. For block codes, error propagation of state-independent decoding is limited to within the span of a codeword. An alternative practical decoding scheme, sliding block decoding [8], bounds error propagation to be within a finite number of codewords.

Generally, a tradeoff among the alternatives described above is needed. For example, to increase the code rate, long codeword lengths in block coding are required [20]. This will result in an increase of complexity in the encoder and decoder, and large error propagation. In addition, as discussed in this thesis, this may also reduce the ability of the code to suppress low-frequency contents around dc.

Given the same order of spectral null at dc, high-order spectral-null multimode codes presented in this thesis show advantages over other proposed high-order spectral-null codes in the literature in the overall consideration of code rate, spectrum performance, codeword length, and implementation.

1.3 Outline of the Thesis

This thesis is organized as follows. In Chapter 2, a formal definition of high-order spectral-null codes is given. Then, alternative representations and characterization of HOSN codes are reviewed. Because most high-order spectral-null codes described in the literature are high-order

6

zero-disparity (HOZD) codes, and since these codes form the basis of the new high-order spectral-null multimode codes introduced in this thesis, some important properties of HOZD codes such as word length and the number of available codewords are investigated. Two fast enumeration algorithms for calculating the cardinality (the number of available words) of second-order zero-disparity codes are developed. An improved algorithm for enumerating the cardinality of HOZD codes is introduced.

Chapter 3 introduces a general performance metric, low-frequency spectrum-weight (LFSW), for arbitrarily high-order spectral-null codes. A review of calculation of the power spectra of block codes with fixed-length and variable-length is followed by the discussion of the previously proposed performance metric for first-order spectral-null codes. Closed-form expressions for LFSW of fixed-length HOSN codes generated through state-dependent and state-independent encoding are derived.

In Chapter 4, coding is categorized to be monomode, bimode and multimode coding. Encoding and decoding of guided scrambling, a multimode coding technique, are considered. Necessary and sufficient conditions for scrambling polynomials to generate complementary quotients in every quotient selection set are derived. Further scrambling polynomials suitable for constructing efficient HOSN multimode codes are obtained through computer search and are given in the following chapter. Dc^1-free multimode codes are also reviewed in Chapter 4.

In Chapter 5, rate-efficient HOSN bimode and multimode codes are discussed. Some properties of those codes are presented. Also in this chapter, the LFSW of HOSN codes with variable length is derived and employed to determine a GS selection criterion that results in good spectrum performance. In addition, another selection criterion for GS is proposed to realize simple implementation of HOSN multimode codes. Power spectrum results of alternative HOSN codes are compared. Scrambling polynomials and selection criteria for generating HOSN multimode codes with high coding efficiency and large suppression of low-frequency components are recommended.

Chapter 6 summarizes this thesis and presents possible future research work related to high-order spectral-null multimode codes.

Chapter 2
HIGH-ORDER SPECTRAL-NULL CODES [2]

To significantly suppress power spectral components at low frequencies, high-order spectral-null (HOSN) codes have been proposed [22], [23]. In this chapter, the definition of HOSN codes is introduced and some characteristics of these codes are reviewed. Further characteristics regarding codeword length and the number of codewords of HOSN codes are developed.

Section 2.1 describes a coding method which can be employed to shape the power spectrum of a coded signal by controlling the correlation of symbols in the coded sequence. It has been shown that high-order running digital sum plays a key role in the existence [26] and rate capacity [33], [25] of the same order spectral-null codes. Definitions of high-order spectral-null codes and high-order running digital sum are introduced in Section 2.2. New characteristics on codeword length and cardinality of high-order zero-disparity codes are developed in Section 2.3 and two fast algorithms for enumerating the number of codewords for second-order spectral-null codes are presented in Section 2.4.

2.1 PSD Shaping of Recording Channel Input Signals

As introduced in Chapter 1, signals stored on the magnetic or optical recording medium are two-level digits. When the relative velocity of a track and the read/write head is constant, each digit along the tracks occupies the same duration T_b and channel digits have a fixed rate $1/T_b$ [4]. The system is called a synchronous storage system. The signals that drive a write head during recording are pulse amplitude modulation (PAM) signals with rectangular pulse shape $s(t)$ of duration T_b, and amplitude values of ± 1 where the channel input discrete sequence $\{x\} = \{x_1, x_2, \cdots, x_l, \cdots\}$ is over the bipolar alphabet $\{-1, +1\}$ and x_l is the amplitude of the PAM signal in the interval $\big((l-1)T_b, lT_b\big]$ [1, p.34], [4]. The PAM signal $X(t)$ is expressed as:

$$X(t) = \sum_{l=1}^{\infty} x_l s(t - (l-1)T_b).$$

(2.1)

In spectral analysis, the first and second order statistics, i.e., the mean and autocorrelation of stochastic processes, are required [34]. Assume that the sequence $\{x\}$ is wide-sense stationary [35]. Then the channel-input signal (PAM signal) is wide-sense cyclostationary with period of T_b [36], [37], [38], i.e.,

---

$$m_X(t) = E\{X(t)\} = m_X(t + T_b) \tag{2.2}$$

$$R_X(t, t+\tau) = E\{X(t)X(t+\tau)\} = R_X(t + T_b, t + T_b + \tau) \tag{2.3}$$

where $E\{\cdot\}$ denotes the statistical average. A cyclostationary process is a special case of a non-stationary stochastic process. Generally, a cyclostationary process is treated as though it were a stationary process by averaging the periodic statistical parameters of the cyclostationary process over one period, or equivalently, by phase-randomizing the cyclostationary process [38]. The power spectrum of the averaged or phase-randomized process is given by [36], [37]:

$$H_X(\omega) = \frac{1}{T_b} |S(\omega)|^2 H_x(\omega) \tag{2.4}$$

where $S(\omega)$ is the Fourier transform of the pulse $s(t)$ and $H_x(\omega)$ is the power spectrum of the sequence $\{x\}^3$ which is the Fourier transform of the autocorrelation, i.e.

$$H_x(\omega) = \sum_{k=-\infty}^{\infty} R_x(k) e^{-ik\omega T_b} = \sum_{k=-\infty}^{\infty} E\{x_l x_{l+k}\} e^{-ik\omega T_b} \tag{2.5}$$

where $i = \sqrt{-1}$. The property of the Fourier transform of discrete-time signals is discussed in [39, Section 4.2.3]. Unlike the Fourier transform of continuous-time signals, the Fourier transform of discrete-time signals is periodic with period $2\pi/T_b$. The frequency components of a discrete-time signal are only unique over the frequency interval $(-\pi/T_b, \pi/T_b)$ or $(0, 2\pi/T_b)$. Therefore, it is enough to evaluate the power spectrum of symbol sequence $H_x(\omega)$ when the normalized frequency ωT_b is within $(-\pi, \pi)$ or $(0, 2\pi)$ (or equivalently the normalized frequency fT_b is within $(-0.5, 0.5)$ or $(0, 1)$). In general, $H_x(\omega)$ is even as shown in Section 3.2.1 and evaluation of $H_x(\omega)$ is considered for ωT_b within $(0, \pi)$ (or for fT_b within $(0, 0.5)$).

Since the channel input signals in conventional recording have the rectangular pulse shape, the power spectra of channel-input signals can be expressed as:

$$H_X(\omega) = T_b \left(\frac{\sin(\omega T_b/2)}{\omega T_b/2} \right)^2 \cdot \sum_{k=-\infty}^{\infty} E\{x_l x_{l+k}\} e^{-ik\omega T_b} . \tag{2.6}$$

Suppressing low-frequency spectral components of recorded signals is accomplished by shaping the power spectrum of the signals. Expressions (2.4) and (2.5) demonstrate that power spectrum shaping of PAM signals $X(t)$ can be realized by designing $S(\omega)$ (equivalently, designing the pulse shape $s(t)$ in the time domain) and by controlling the statistical correlation

---

[3] Calculation of power spectra of symbol sequences will be considered in detail in Chapter 3 of this thesis.

of symbols in the sequence $\{x\}$. However, for conventional magnetic or optical recording channels, the ability to modify the standard pulse shape of channel input signals is limited [1, p.34]. In this case, the power spectrum of the sequence $\{x\}$ must be shaped in order to shape the power spectrum of signals $X(t)$.

There are several alternative ways of controlling the power spectrum of transmitted signals in baseband digital transmission systems including techniques such as line coding, filtering, etc [40]. Among these techniques, line coding is the primary method used in optical and metallic cable transmission systems [16], [41], [42]. Similar to line coding techniques in digital transmission systems, modulation coding is used in recording systems for control of the power spectrum of recorded symbol sequences. The difference between line coding and modulation coding is not in their coding mechanisms, but in the applied systems. As described in Chapter 1, first-order spectral-null codes, which are widely used modulation codes in digital recording systems, have the characteristic that the power spectrum of the coded sequence $\{x\}$, $H_x(\omega)$, has a spectral null at dc. To realize larger rejection of low-frequency spectral contents around dc, a deeper spectral null at dc is needed. High-order spectral-null codes are employed for this purpose. In the following section, the definition and characterization of high-order spectral codes are introduced.

2.2 Sequences with a *K*th-Order Spectral Null at Dc

2.2.1 Definition

Let the sequence $\{x\} = \{x_1, x_2, \cdots, x_l, \cdots\}$ be a dc-free encoded symbol sequence over a finite alphabet $\mathcal{I}$. In the remainder of this thesis, for convenience, symbols "+1" and "−1" in the bipolar alphabet $\{-1, +1\}$ and in the ternary alphabet $\{-1, 0, +1\}$ will be represented as " + " and " − "; binary symbols will be represented from the alphabet $\{0, 1\}$. The bipolar version of binary symbols is obtained by substituting the symbol −1 for the symbol 0. The running digital sum (RDS) is defined as the accumulation of the symbol values in a sequence starting from the beginning of the sequence to the time instant of interest. The necessary and sufficient condition for the existence of a dc^1-free code is that the RDS values of the coded sequence measured at any time instant are finite [43].

For a *K*th-order spectral-null code $(K \geq 1)$, the power spectrum of the encoded sequence and up to its $2K - 1$ derivatives are zero at dc [27]. Such a code is denoted to be a dc^K-free code.

Consider the encoded symbol sequence $\{x\} = \{x_1, x_2, \cdots, x_l, \cdots\}$. Define $\mathrm{RDS}^{(r)}(l)$ to be the rth-order RDS of $\{x\}$ at discrete-time instant l satisfying:

$$\mathrm{RDS}_l^{(r)} = \mathrm{RDS}_{l-1}^{(r)} + \mathrm{RDS}_l^{(r-1)}, \tag{2.7}$$

where $l \geq 1$ and $r \geq 1$. $\mathrm{RDS}_0^{(r)}$ represents the initial value of the rth-order RDS of the sequence. Note that $\mathrm{RDS}_l^{(0)} = x_l$. The necessary and sufficient condition for a dc^K-free code is that its Kth-order running digital sum is bounded [25], [26]. If the highest order RDS is bounded, all lower order RDS values are bounded [25].

Dc^K-free codes typically are generated through block encoding [26], where the source symbol sequence is framed into a series of words of symbols, and the source words are mapped to words of coded symbols according the coding rule. Block decoding reverses this procedure. Usually, the source word and codeword have a fixed length [34]. Denote fixed-length dc^K-free codes with length-M binary source words and length-N bipolar codewords to be M/N dc^K-free codes, and block dc^K-free ternary codes with length-M binary source words and length-N ternary codewords to be MB/NT dc^K-free codes. In all cases, M and N are positive integers.

Modulation/line codes with variable lengths of source words and codewords have also received attention [1, ch.7.2], [44]. Practical variable-length codes are so called *synchronous variable-length codes* where the quotient of the lengths of a source word and its corresponding codeword is constant.

Figure 2.1 shows the power spectra of two bipolar codes, the well-known Manchester code, which is a $1/2$ dc^1-free code and is also known as the bi-phase code in which the binary source symbols "0" and "1" are encoded to be bipolar symbols "$-+$" and "$+-$" respectively, and the $2/4$ dc^2-free code constructed in [22].

Figure 2.2 shows the power spectra of two ternary codes, the widely used alternate mark inversion (AMI) code [40] which is $1B/1T$ dc^1-free code, and the $3B/3T$ dc^2-free code [26].

Since in this thesis it is of interest to consider suppression of low-frequency spectral components of coded sequences, generally only the low-frequency part of the spectrum is illustrated rather than the entire spectrum of the coded sequence. Figures 2.1 and 2.2 demonstrate that it is possible to suppress low-frequency spectral components of codes with the same code rate but with a higher order spectral null at dc. The cost for implementation of such a higher-order spectral null code is usually increased codeword length and complexity of coding.
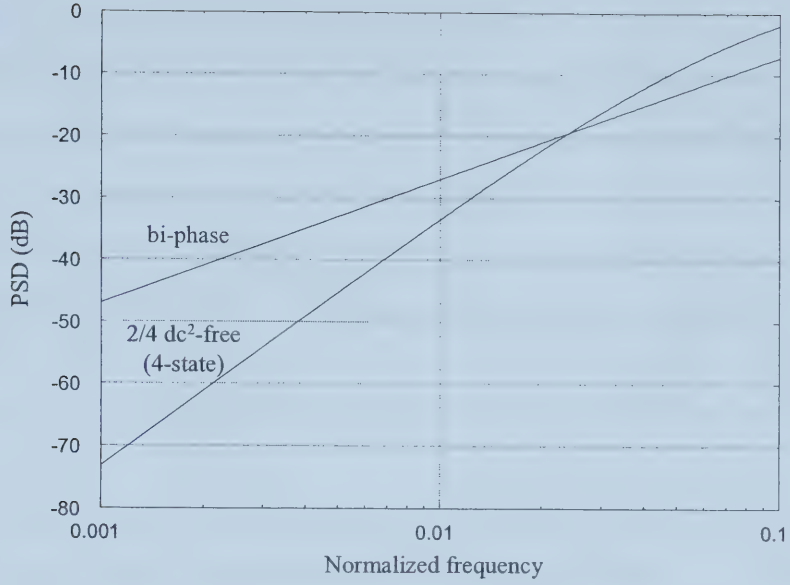
Figure 2.1 Power spectra of the bi-phase code and the dc²-free code (4-state).
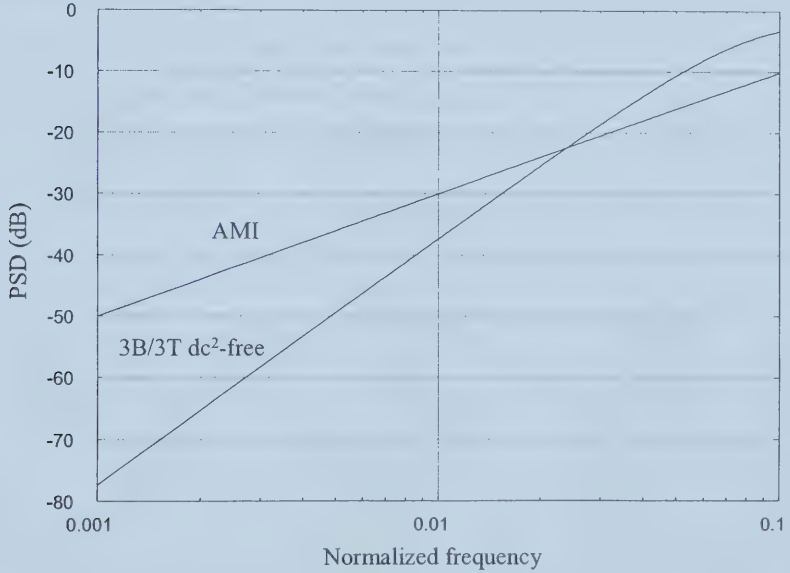
Figure 2.2 Power spectra of the AMI code and the 3B/3T dc²-free code.

2.2.2   Finite-State Sequential Machine Model of a Dc^K-Free Encoder

An encoder generating sequences satisfying given constraints can be modeled as a finite-state sequential machine (FSSM) [45]. A symbol-by-symbol encoder is described by the quintuple $\mathcal{E} = (\mathbb{Y}, \mathbb{X}, \mathcal{S}, g, h)$ where $\mathbb{Y} \subseteq \mathcal{I}$ is the set of input symbols y_i; $\mathbb{X} \subseteq \mathcal{I}$ is the set of output symbols x_i; $\mathcal{S} = \{\sigma_1, \cdots, \sigma_S\}$ is the set of encoder states and S is the number of states; g is the

12

state transition function; and h is the output function [43]. The state sequence $\{s_l\}$ ($s_l \in S$ is the state at time instant l) is a Markov chain where the next state $s_{l+1} = g(s_l, y_l)$ is related to the present state s_l and the present input symbol y_l. Both input and output sequences of the encoder, $\{y\}$ and $\{x\}$, are stochastic processes. Assuming that the input source sequence $\{y\}$ is composed of independent identically distributed (i.i.d.) symbols, the state sequence $\{s_l\}$ is a stationary Markov chain [43] and the output symbols of the FSSM are a memoryless function of this stationary Markov chain. Therefore, $x_l = h(s_l)$ for a Moore state machine model, or equivalently, $x_l = h(s_l, y_l)$ for a Mealy state machine model [34]. The output sequence $\{x\}$ is stationary [46].

The finite-state transition diagram (FSTD) [8], a graphic representation of an encoder, is also very useful for analysis and construction of a dc^K-free code [24]-[26]. An FSTD generating sequences with a Kth-order spectral null at dc, called a dc^K-FSTD, is a directed graph in which the nodes represent the states in S (each state is represented by the RDS$^{(r)}$ values $r = 1, \cdots, K$), the directed edges connecting nodes express state transitions, and the labels on the edges are the output symbols of the associated departure states for the FSSM.

Figure 2.3 is a dc^1-FSTD with the bipolar alphabet. The states in this figure are represented by the RDS$^{(1)}$ values that are within $[V_-, V_+]$, where V_- and V_+ are finite.
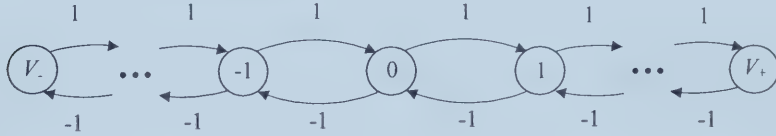
Figure 2.3 Dc^1-FSTD with bipolar alphabet.

Figure 2.4 shows a dc^2-FSTD with the bipolar alphabet. In this figure, states are determined by both RDS$^{(1)}$ and RDS$^{(2)}$ values, and the absolute RDS$^{(1)}$ and RDS$^{(2)}$ values are not larger than two. In this diagram the solid and dashed lines represent the encoder outputs being $+1$ and -1 respectively. All state transitions of sequences with a second order spectral null at zero frequency and with absolute RDS$^{(2)}$ values not larger than two are within the diagram in Figure 2.4.
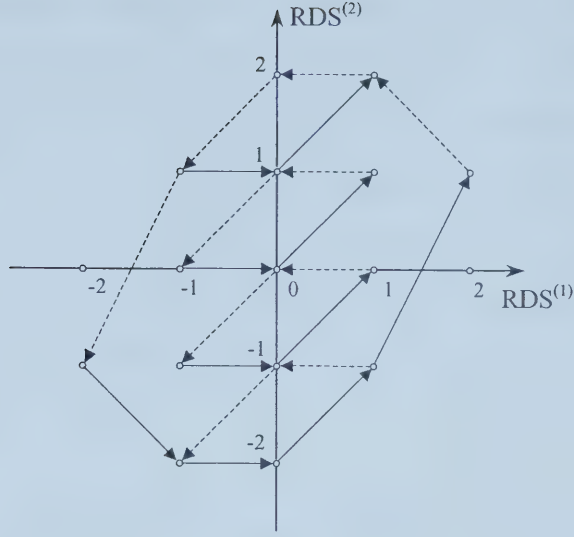
Figure 2.4 Dc^2-FSTD with bipolar alphabet.

Although symbol-by-symbol encoding through the dc^K-FSTD described above can generate sequences with a Kth-order spectral null at zero frequency, in the corresponding decoder, the same trellis diagram is also needed for decoding. This is called state-dependent decoding. However, a single channel bit error at the input to such a state-dependent decoder can result in unbounded error propagation that could corrupt the entire decoded data sequence [4], [8]. To avoid such error propagation, a decoder is preferably state-independent in practice [4]. It is desirable that in a decoder, errors in its input sequence lead to only a finite number of errors at its output. To do so, the decoding should be considered within a finite observation interval of channel bits. Currently, there are two kinds of practical decoders for constrained systems: one is the widely used *block decoder* [4], [16]; the other is the so-called *sliding-block decoder* [4], [8], [24], [47].

Both the block decoder and the sliding-block decoder require block encoding which is described in Section 2.2.1. Let the source word length and codeword length be M and N respectively. In a block decoder, a length-N input word is mapped to a length-M decoded word based on the decoding rule. Error propagation in block decoding is limited to within a decoded word. The smaller the value of M, the less the error propagation. Throughout this thesis, block decoding is used exclusively for the considered high-order spectral-null multimode codes. Alternatively, in a sliding-block decoder, a length-M decoded word is determined not only by the corresponding received length-N word but by p preceding words and u upcoming words, where p and u are finite non-negative integers. Error propagation of a sliding-block decoder is limited to within $p+u+1$ words [8].

Similar to a symbol-by-symbol encoder, a block encoder can also be defined as a quintuple $\mathcal{E} = (\mathbb{Y}^M, \mathbb{X}^N, \mathcal{S}, g, h)$ where $\mathbb{Y}^M \subseteq \mathcal{I}^M$ is the set of input M-symbol row-vector-valued words $\underline{y}_n = (y_{n,1}, \cdots, y_{n,M})$, $M \geq 1$; $\mathbb{X}^N \subseteq \mathcal{I}^N$ is the set of output N-symbol row-vector-valued words $\underline{x}_n = (x_{n,1}, \cdots, x_{n,N})$, $N \geq 1$; $\mathcal{S}, g, h$ are as defined for the symbol-by-symbol decoder and each state is assumed to be determined by the RDS$^{(r)}$ $(r = 1, \cdots, K)$ values at the end of theoutput words. The main difference is that the input and output of the Markov process $\{s_n\} \in \mathcal{S}$ are vector valued, that is $s_{n+1} = g(s_n, \underline{y}_n)$ and $x_n = h(s_n)$ or $\underline{x}_n = h(s_n, \underline{y}_n)$ during the nth encoding interval.

2.2.3   Capacity

For a binary discrete noiseless input-constrained channel, Shannon first defined the capacity C for the channel and proved that this capacity represents an upper bound on the achievable rate of information transmission on the channel [32]. The capacity C is expressed as:

$$C = \log_2 \lambda_{\max} \qquad \text{bit / symbol,} \tag{2.8}$$

where $\lambda_{\max}$ is the largest eigenvalue of the state transition connection matrix of the state sequence of the Markov chain in FSSM described above where the symbol "1" at the ith row and jth column in this connection matrix means the existence of one-step transition from the ith state to the jth state; otherwise the symbol "0" in this matrix means no such a transition.

A.   Capacity of a dc^1-free constrained channel

Let the running digital sum variation be $V = V_+ - V_- + 1$ where V_+ and V_- are as defined in Figure 2.3. The adjacent $V \times V$ matrix A of the state sequence follows from Figure 2.3:

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & \cdots & 0 \\ 1 & 0 & 1 & 0 & \cdots & 0 \\ 0 & 1 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & 0 & 1 \\ 0 & 0 & \cdots & 0 & 1 & 0 \end{pmatrix}.$$

By evaluating the largest eigenvalue of this matrix, a closed-form expression for the capacity of a bipolar dc^1-free constrained channel is obtained, and is related to the given running digital variation value [33]:

$$C(V) = 1 + \log_2 \cos \frac{\pi}{V+1}, \qquad V \geq 3. \tag{2.9}$$

B.   Capacity of dc²-free and dc³-free constrained channels

Unlike the case of dc^1-free constrained channels, to date closed-form expressions for capacities of higher-order spectral-null constrained channels have not yet been derived. Some numerical results for capacities of dc^2-free and dc^3-free constrained channels are given in [25]. It has also been shown [25] that the capacities of dc^2-free and dc^3-free constrained channels are dependent only on the variation of RDS$^{(2)}$ and RDS$^{(3)}$ values respectively. In Section 3.3 of this thesis, it will be shown that the asymptotic power spectrum of a dc^K-free code is also related to the statistics of RDS$^{(K)}$ values of the coded sequence.

The simplest coding method for generating sequences with a high-order spectral null at zero frequency is to employ a state-independent block code. These types of codes are discussed in the following section. Improved lower bounds for word length and redundancy of these codes and new algorithms for enumerating the number of codewords in these codes are derived.

2.3 *K*th-Order Zero-Disparity (*K*-OZD) Codes

2.3.1 Alternative Representations

Let a word be represented by a vector $\underline{x} = (x_1, x_2, \cdots, x_N) \in \{-1, +1\}^N$. Define a *z-polynomial* $X(z)$ as below:

$$X(z) = x_1 z + x_2 z^2 + \cdots + x_N z^N .$$ (2.10)

Then the discrete Fourier transform of $\underline{x}$ follows by simply replacing $z = e^{-i\omega}$ into $X(z)$ where $i = \sqrt{-1}$, i.e.

$$X(e^{-i\omega}) = X(z)\big|_{z=e^{-i\omega}} = \sum_{j=1}^{N} x_j e^{-ij\omega} .$$ (2.11)

The word $\underline{x}$ is defined to be a dc^K-free word [23], [27] if

$$\frac{d^r \left| X(e^{-i\omega}) \right|^2}{d\omega^r} \Bigg|_{\omega=0} = 0 , \quad \text{for } r = 0, 1, \cdots, 2K - 1.$$ (2.12)

Construction of dc^K-free codes is typically based on free concatenation of dc^K-free words, that is, state-independent encoding [23], [27], [28], [30]. This class of M/N dc^K-free codes is denoted to be M/N Kth-order zero-disparity (K-OZD) codes in which there exist one-to-one mappings between source words and codewords. The disparity of a word is defined as the difference between the number of logic ones and the number of logic zeros.

It can be verified [1, p. 243] that if $X(e^{-i\omega})$ satisfies:

$$\frac{d^r X(e^{-i\omega})}{d\omega^r}\bigg|_{\omega=0} = 0, \quad \text{for } r = 0, 1, \cdots, K-1 \tag{2.13}$$

or equivalently,

$$\frac{d^r X(z)}{dz^r}\bigg|_{z=1} = 0, \quad \text{for } r = 0, 1, \cdots, K-1, \tag{2.14}$$

the word $\underline{x}$ is a dc^K-free word [27]. Denote the set of all dc^K-free words as $\mathcal{A}(N, K) \subseteq \{-1, +1\}^N$ where N is the word length and K is the order of spectral null at dc. Three equivalent representations of dc^K-free words are presented below.

A. *Moment representation* [23], [27]

The rth-order moment (rth-order disparity) of a word $\underline{x} = (x_1, x_2, \cdots, x_N) \in \{-1, +1\}^N$ $(r \geq 0)$, $u^{(r)}$, is defined as $u^{(r)} = \sum_{j=1}^{N} j^r x_j$. The word $\underline{x}$ is a dc^K-free word, if

$$u^{(r)} = 0, \quad \text{for } r = 0, 1, \cdots, K-1. \tag{2.15}$$

B. *Polynomial representation* [27], [29]

From (2.14), it is straightforward to verify that the z-polynomial $X(z)$ of a dc^K-free word $\underline{x}$ has a factor of $(z-1)^K$. Therefore, $X(z)$ represents a dc^K-free word if, and only if, $(z-1)^K | X(z)$.

C. *Number theory representation*

Dc^K-free words were first associated with the Prouhet-Tarry-Escott problem in number theory [48]-[50] in [23], resulting in the fact that analysis developed for the Prouhet-Tarry-Escott problem can conveniently be applied to analysis of Kth-order zero-disparity codes. Let $\alpha = \{\alpha_1, \cdots, \alpha_{N/2}\}$ and $\beta = \{\beta_1, \cdots, \beta_{N/2}\}$ be the two disjoint sets of indices of the word $\underline{x}$ in which $x_{\alpha_p} = +1$ and $x_{\beta_q} = -1$, where $\alpha_p, \beta_q \in \{1, \cdots, N\}$ and $p, q \in \{1, \cdots, N/2\}$. From (2.15), if $\underline{x}$ is a dc^K-free word, then:

$$\sum_{p=1}^{N/2} \alpha_p^r = \sum_{q=1}^{N/2} \beta_q^r, \quad \text{for } r = 0, 1, \cdots, K-1. \tag{2.16}$$

Other representations of dc^K-free words have been given in [27], [50].

There are still many challenges in the theory of high-order zero-disparity codes, some of which were overviewed in [29]. Knowledge of word length and the number of available words of dc^K-free words is important in construction of a block K-OZD code. This will be discussed in the remainder of this chapter.

2.3.2 Word Length

A. Factorization of word length

It is straightforward to observe that a dc^1-free word in $\mathcal{A}(N,1)$ has equal numbers of "$+$"s and "$-$"s, and that this is possible only if N is even. Therefore $\mathcal{A}(N,1) \neq \varnothing$ if, and only if, N is even [51]. It has been shown that $\mathcal{A}(N,2) \neq \varnothing$ only if N is divisible by 4 [23]. Theorem 2.1 below is a new result that shows the sufficiency of the word length for $\mathcal{A}(N,2)$.

Theorem 2.1: If N is a multiple of 4, $\mathcal{A}(N,2) \neq \varnothing$.

Proof: Assume that the word length N is divisible by 4. Then $N = 2s$, where s is even. Let α_p and β_q be indices of the word $\underline{x}$ in which $x_{\alpha_p} = +1$ and $x_{\beta_q} = -1$ where $\alpha_p, \beta_q \in \{1, \cdots, N\}$. From the definition of moment of words, when $u^{(0)} = 0$, $p,q \in \{1, \cdots, N/2\}$. Since $\alpha_p \neq \beta_q$, then

$$\alpha_1 + \cdots + \alpha_s + \beta_1 + \cdots + \beta_s = 1 + 2 + \cdots + N = (1/2)N(N+1) = s(2s+1).$$

Also

$$1 + \cdots + s \leq \alpha_1 + \cdots + \alpha_s \leq (s+1) + \cdots + 2s,$$

and the summation $\alpha_1 + \cdots + \alpha_s$ can be any integer value in the range $[(1/2)s(s+1), (1/2)s(3s+1)]$. There must exist words such that

$$\alpha_1 + \cdots + \alpha_s = (1/2)s(2s+1) = \beta_1 + \cdots + \beta_s$$

implying the existence of words in $\mathcal{A}(N,2)$ [27]. □

A general condition regarding the necessity of word length for $\mathcal{A}(N,K)$ has been developed and is given by:

Theorem 2.2 [27],[29],[52]: The set $\mathcal{A}(N,K)$ is empty if N is not divisible by $2^{\lfloor \log_2 K \rfloor + 1}$.

There is a historic result on the Prouhet-Tarry-Escott problem in number theory which was overviewed in [53, Theorem 7]. This result can be applied to the theory of high-order zero-disparity codes. It has been shown that:

18

Theorem 2.3: For the word length $N = 2^{\mathcal{K}}(2m+1)$, if $m=0$, $\mathcal{A}(N,\mathcal{K}) \neq \varnothing$; if $m>0$, $\mathcal{A}(N, \mathcal{K}+1) \neq \varnothing$.

B. *Minimum word length*

The problem of minimum word length of dc^K-free words has attracted interest in both mathematics and engineering [27], [52], [54], [55]. Let $N^*(K)$ denote the minimum word length of dc^K-free words. Let $\mathcal{P}(N)$ denote the set of polynomials with all coefficients over $\{-1,1\}$ and with degree $N-1$ (the coefficient of the highest degree is 1). Let $\mathcal{P}(N,K)$ denote the subset of the set $\mathcal{P}(N)$ in which an element $X(z) \in \mathcal{P}(N,K)$ is divisible by $(z-1)^K$ (or by a higher power of $z-1$). Also let $A_N(z) = 1 + z + \cdots + z^{N-1} = (z^N - 1)/(z-1)$. If $X(z) \in \mathcal{P}(N)$, then $X(z) \equiv A_N(z) \pmod{2}$. Byrnes raised a conjecture that $N^*(K) = 2^K$ and stated that "a proof that the degree of such a X must be exponential in K would be of interest" [54]. It has been shown through enumeration that for $K \leq 5$, the conjecture is true and the available polynomial $X(z)$ is unique [23], [52]. Furthermore, Boyd disproved the conjecture for the case of $K \geq 6$ [52] and proved that $N^*(6) = 48$ [52] and $N^*(7) = 96$ [55]. The answer to $N^*(K)$ ($K \geq 8$) is still open. Searching available polynomials in $\mathcal{P}(N,K)$ by computer is a great challenge for large values of N and K such as $N = 96$ and $K = 8$ [55].

From Theorem 2.3, it is straightforward to show that:

$$N^*(K) \leq 2^K. \tag{2.17}$$

Boyd introduced a lower bound on $N^*(K)$ given by Theorem 2.4 below.

Theorem 2.4 [52, Theorem 1]: If $\mathcal{P}(N,K)$ is nonempty and if p is a prime which does not divide N, then

$$N^*(K) \geq \left(p^{1/(p-1)}\right)^K. \tag{2.18}$$

It is observed that Boyd's lower bound on $N^*(K)$ shown in (2.18) is obtained under a "loose" assumption. By reconsidering the assumption previously given, an improved lower bound on $N^*(K)$ can be obtained and is given below in Theorem 2.5. Some preliminaries are required in order to develop this theorem.

Let p be a prime and let ξ_p denote a pth root of unity [56], i.e., $\xi_p = e^{i2\pi/p}$ where $i = \sqrt{-1}$. Then, $\xi_p^j = e^{i2\pi j/p}$, $j = 1,\ldots,p-1$ are all roots of equation $A_p(z) = 1 + z + \cdots + z^{p-1} = 0$.

Lemma 2.1: p non-negative consecutive powers of ξ_p^j form all pth roots of unity.

Proof: Let p non-negative consecutive powers of ξ_p^j be $(\xi_p^j)^{t_0+t}$ where t_0 is a non-negative integer and $t = 0, 1, \ldots, p-1$. Then:

$$(\xi_p^j)^{t_0+t} = (e^{i2\pi j/p})^{t_0+t} = e^{i2\pi jt_0/p} \cdot e^{i2\pi jt/p}, \quad j = 1, \ldots, p-1 \text{ and } t = 0, 1, \ldots, p-1. \tag{2.19}$$

It is straightforward to verify that $e^{i2\pi x/p} = e^{i2\pi(x \bmod p)/p}$ where x is a non-negative integer. According to Fermat's Theorem [57, p.39], all p numbers $0 \bmod p, 1 j \bmod p, \cdots,$ $(p-1) j \bmod p$ are distinct. It can also be shown that p numbers $(0 + jt_0) \bmod p,$ $(1 + jt_0) \bmod p, \cdots, ((p-1) + jt_0) \bmod p$ are all distinct, from which the lemma follows. $\square$

Lemma 2.2 [52, Corollary 1]: If $X(z) \in \mathcal{P}(N)$ and p does not divide N, then $X(\xi_p^j) \neq 0$.

Theorem 2.5: If $\mathcal{P}(N, K)$ exists and if p is a prime that does not divide N, then

$$N^*(K) \geq (p/\Delta)(p^{K/(p-1)} - p + 1), \tag{2.20}$$

where Δ is a real number less than p.

Proof: If $X(z) \in \mathcal{P}(N, K)$, then

$$X(z) = (z-1)^K Y(z) \tag{2.21}$$

where $Y(z)$ has integer coefficients. Replacing ξ_p^j ($j = 1, \ldots, p-1$) into (2.21), since p does not divide N, from Lemma 2.2 there is:

$$0 \neq X(\xi_p^j) = (\xi_p^j - 1)^K Y(\xi_p^j), \quad j = 1, \ldots, p-1. \tag{2.22}$$

The absolute Norm $X(\xi_p) = \prod_{j=1}^{p-1} X(\xi_p^j)$ is:

$$\left| \text{Norm} X(\xi_p) \right| = \left| \prod_{j=1}^{p-1} X(\xi_p^j) \right| = \left| \prod_{j=1}^{p-1} (\xi_p^j - 1)^K Y(\xi_p^j) \right| = \left| \prod_{j=1}^{p-1} (\xi_p^j - 1)^K \right| \cdot \left| \prod_{j=1}^{p-1} Y(\xi_p^j) \right|. \tag{2.23}$$

For an arbitrary root ξ_p^j of $A_p(z)$, from Lemma 2.1, it can be shown that:

$$\left| X(\xi_p^j) \right| = \left| 1 \pm \xi_p^j \pm (\xi_p^j)^2 \pm \cdots \pm (\xi_p^j)^{N-1} \right|$$

$$\leq \left| 1 \pm \xi_p^j \pm \cdots \pm (\xi_p^j)^{p-1} \right| + \left| (\xi_p^j)^p \pm (\xi_p^j)^{p+1} \pm \cdots \pm (\xi_p^j)^{2p-1} \right| + \cdots +$$

$$\left| (\xi_p^j)^{\lfloor N/p \rfloor p} \pm (\xi_p^j)^{(\lfloor N/p \rfloor p+1)} \pm \cdots \pm (\xi_p^j)^{(N-1)} \right|$$

$$= \underbrace{\left| 1 \pm \xi_p^1 \pm \cdots \pm \xi_p^{p-1} \right| + \cdots + \left| 1 \pm \xi_p^1 \pm \cdots \pm \xi_p^{p-1} \right|}_{\lfloor N/p \rfloor} + \underbrace{\left| (\xi_p^j)^{\lfloor N/p \rfloor p} \pm (\xi_p^j)^{(\lfloor N/p \rfloor p+1)} \pm \cdots \pm (\xi_p^j)^{(N-1)} \right|}_{\leq p-1}.$$

If $\left| 1 \pm \xi_p^1 \pm \cdots \pm \xi_p^{p-1} \right| \leq \Delta$ where $\Delta < p$, then

20

$$\left|X(\xi_p^j)\right| \le \Delta\lfloor N/p \rfloor + \underbrace{\left|(\xi_p^j)^{\lfloor N/p \rfloor p} \pm \cdots \pm (\xi_p^j)^{N-1}\right|}_{\le p-1} \le \Delta N/p + p - 1 \tag{2.24}$$

and

$$\left|\prod_{j=1}^{p-1} X(\xi_p^j)\right| \le (\Delta N/p + p - 1)^{p-1}. \tag{2.25}$$

Since ξ_p^j is a root of polynomial $A_p(z)$, $A_p(z) = 1 + z + \cdots + z^{p-1} = \prod_{j=1}^{p-1}(z - \xi_p^j)$. It is

straightforward to show that:

$$\left|\prod_{j=1}^{p-1}(\xi_p^j - 1)^K\right| = \left(A_p(1)\right)^K = p^K. \tag{2.26}$$

As estimated in [52],

$$\left|\prod_{j=1}^{p-1} Y(\xi_p^j)\right| \ge 1. \tag{2.27}$$

By combining (2.25)-(2.27), the theorem follows. $\qquad\qquad\square$

Example 2.1: Let $p = 3$. The 3rd root of unity is $\xi_3 = e^{i2\pi/3}$. Due to the unknown signs in

the inequality $\left|1 \pm \xi_p^1 \pm \cdots \pm \xi_p^{p-1}\right| \le \Delta$, to determine the value of Δ an enumerative method is

employed as shown below:

$$\left|1 + \xi_3^1 + \xi_3^2\right| = 0 ;$$
$$\left|1 - \xi_3^1 + \xi_3^2\right| = \left|1 + \xi_3^1 - \xi_3^2\right| = 2 ;$$
$$\left|1 - \xi_3^1 - \xi_3^2\right| = 2.$$

So $\left|1 \pm \xi_3^1 \pm \xi_3^2\right| \le 2$, or $\Delta = 2$. Using the same method, it can be shown that for $p = 5$,

$\Delta = 3.236$. $\qquad\qquad\Diamond$

Corollary 2.1: If $\mathcal{P}(N, K)$ exists and if p is a prime that is not divisible by N, then

a) if $p = 3$, $N^*(K) \ge 1.5(3^{K/2} - 2)$;

b) if $p = 5$, $N^*(K) \ge (5/3.236)(5^{K/4} - 4)$.

Determination of Δ requires enumeration and evaluation of 2^{p-1} magnitude values. When p

is relatively large, this enumeration becomes prohibitive. The following conjecture regarding the

value of Δ for a given prime p is based on Lemma 2.3 and the observation below.

Lemma 2.3: Let p be an odd prime. For two arbitrary adjacent roots out of all pth roots of unity, the bisector of these two roots on the unit circle is the reverse of one of the other pth roots of unity.

Proof: Let two adjacent roots be $e^{i2\pi j/p}$ and $e^{i2\pi(j+1)/p}$ where $0 \le j \le p-1$ and $i = \sqrt{-1}$. The bisector of these two roots is $e^{i2\pi(2j+1)/2p}$ and the reverse this bisector is $-e^{i2\pi(2j+1)/2p} = e^{i\pi(2j+p+1)/p}$. Since p is an odd prime, it can be denoted $p = 2k+1$. The reverse of the bisector of roots is $e^{i2\pi(k+j+1)/p} = e^{i2\pi j^*/p}$, where $0 \le j^* \le p$. When $k+j+1 < p$, $j^* = k+j+1$; otherwise, $j^* = k+j+1-p$. □

From Lemma 2.3 above, it is straightforward to see that, for all pth roots of unity, if the roots in Quadrant II and III on the complex plane are mapped to Quadrant I and IV, then on the right half of the unit circle, two adjacent complex numbers of p complex numbers are separated by π/p, and the conjugate of each number exists. Figure 2.5 illustrates an example of these mappings for $p = 7$.
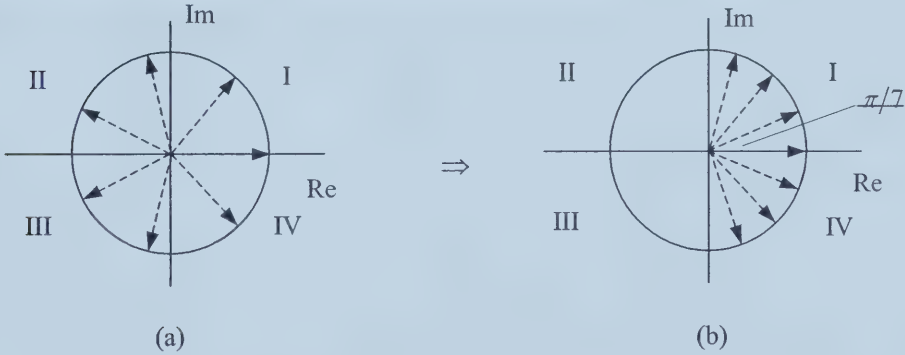
Figure 2.5 Illustration of mappings of all pth roots of unity for $p = 7$.

Denote the p complex numbers after the mappings described above to be the densified pth roots of unity. Let the sum of all densified pth roots of unity be Z. Then:

$$|Z| = \left| \sum_{j=-(p-1)/2}^{(p-1)/2} e^{i\pi j/p} \right| = 1 + 2 \sum_{j=1}^{(p-1)/2} \cos(j\pi/p). \tag{2.28}$$

From [58, p. 35], it can be shown that:

$$|Z| = 1 + \frac{2\cos((p+1)\pi/4p)\sin((p-1)\pi/4p)}{\sin(\pi/2p)}$$

$$= \frac{1}{\sin(\pi/2p)}. \tag{2.29}$$

22

Note that when the complex plane rotates θ_0, $0 \le \theta_0 \le 2\pi$,

$$\left| Z_{\theta_0} \right| = \left| \sum_{j=-(p-1)/2}^{(p-1)/2} e^{i(j\pi/p+\theta_0)} \right| = \left| \sum_{j=-(p-1)/2}^{(p-1)/2} e^{ij\pi/p} \right| \left| e^{i\theta_0} \right| = \left| Z \right|$$

implying that the magnitude of the sum of rotated densified pth roots of unity has the same value.

For the cases of $p = 3$ and 5, it has been observed that:

$$\left| 1 \pm \xi_p^1 \pm \cdots \pm \xi_p^{p-1} \right| = \left| 1 \pm e^{i2\pi/p} \pm \cdots \pm e^{i2\pi(p-1)/p} \right| \le \left| \sum_{j=-(p-1)/2}^{(p-1)/2} e^{i(j\pi/p+j_0\pi/p)} \right| = \frac{1}{\sin(\pi/2p)},$$

where $j_0 = 0, 1, \cdots, 2p-1$.

Conjecture 2.1: For an arbitrary odd prime p, ξ_p is the pth root of unity. Therefore

$$\left| 1 \pm \xi_p^1 \pm \cdots \pm \xi_p^{p-1} \right| \le \Delta = \frac{1}{\sin(\pi/2p)}.$$

As noted in [52], it is desirable that the number p in Theorems 2.4 and 2.5 is chosen to be the smallest prime not dividing N. Figure 2.6 shows the true values of $N^*(K)$ known to date, Boyd's lower bound on $N^*(K)$ given by Theorem 2.4, the new lower bound on $N^*(K)$ given by Theorem 2.5, and the upper bound on $N^*(K)$ based on (2.17). Assume that p equals 3.
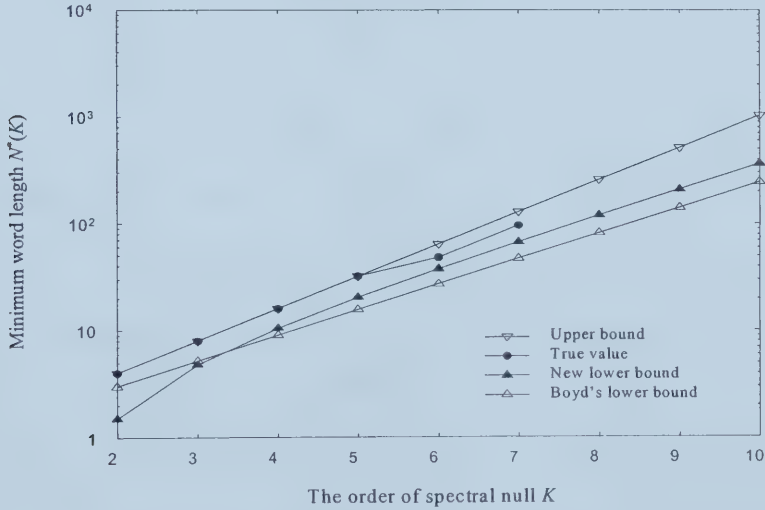
Figure 2.6 Comparison of true values and bounds for minimum word length $N^*(K)$.

2.3.3 Redundancy

In construction of Kth-order zero disparity codes, given the word length N and the order of spectral null K, the code rate is exclusively determined by the size of the codes, i.e., $\left| A(N, K) \right|$.

Assume that the encoder has binary inputs and that the source word length is M. Practical K-OZD codes require that $|\mathcal{A}(N, K)| \geq 2^M$. Enumerating the exact number of codewords in $\mathcal{A}(N, K)$ and estimating the size of K-OZD codes are also challenges in the theory of K-OZD codes for large values of N and K [29], [52], [55].

It is straightforward to verify that $|\mathcal{A}(N, 1)| = C_{N/2}^N$. A general method for enumerating the exact number of words in $\mathcal{A}(N, K)$ was presented and the results for relatively small values of N and K ($N \leq 32$, $K \leq 5$) were given in [23]. This enumeration algorithm is reconsidered below with new initial conditions resulting in further results that are listed in Table 2.1. Let $\underline{x}_a = [x_{a,1}, x_{a,2}, \cdots, x_{a,N}]$ ($x_{n,j} \in \{-1, +1\}$, $j = 1, \cdots, N$) be an element in the set $\mathcal{A}(N, K)$. From (2.15), the rth-order moment of the word $\underline{x}_a$, $u_a^{(r)}$, satisfies:

$$u_a^{(r)} = 0, \quad \text{for } r = 0, 1, \cdots, K-1. \tag{2.30}$$

Also let $\alpha = \{\alpha_1, \cdots, \alpha_{N/2}\}$ and $\beta = \{\beta_1, \cdots, \beta_{N/2}\}$ be as defined in Section 2.3.1. Let $J_r = (1/2)\sum_{j=1}^{N} j^r$, $r = 0, 1, \cdots, K-1$. Then:

$$\sum_{p=1}^{N/2} \alpha_p^r + \sum_{q=1}^{N/2} \beta_q^r = \sum_{j=1}^{N} j^r = 2J_r, \tag{2.31}$$

and

$$\sum_{p=1}^{N/2} \alpha_p^r - \sum_{q=1}^{N/2} \beta_q^r = u_a^{(r)}, \quad r = 0, 1, \cdots, K-1. \tag{2.32}$$

Since (2.31) holds for the dc^K-free word $\underline{x}_a$, there is:

$$\sum_{p=1}^{N/2} \alpha_p^r = \sum_{q=1}^{N/2} \beta_q^r = (1/2)\sum_{j=1}^{N} j^r = J_r, r = 0, 1, \cdots, K-1. \tag{2.33}$$

A K-variable generating function can be used to enumerate the number of dc^K-free words [23]. Define the enumerating generating function to be:

$$F_N(v_0, v_1, \cdots, v_{K-1}) = \prod_{j=1}^{N} (1 + v_0^{j^0} v_1^{j^1} \cdots v_{K-1}^{j^{(K-1)}})$$

$$= \sum_{\lambda_0, \lambda_1, \cdots, \lambda_{K-1}} \Lambda_N(\lambda_0, \lambda_1, \cdots, \lambda_{K-1}) \cdot v_0^{\lambda_0} v_1^{\lambda_1} \cdots v_{K-1}^{\lambda_{K-1}}. \tag{2.34}$$

It can be shown that when $\lambda_r = J_r, r = 0, 1, \cdots, K-1$, a valid codeword $\underline{x}_a$ is enumerated. From (2.34), it is straightforward to obtain the recursive relation below:

$$\Lambda_n(\lambda_0, \lambda_1, \cdots, \lambda_{K-1}) = \Lambda_{n-1}(\lambda_0, \lambda_1, \cdots, \lambda_{K-1}) + \Lambda_{n-1}(\lambda_0 - n^0, \lambda_1 - n^1, \cdots, \lambda_{K-1} - n^{(K-1)}), \tag{2.35}$$

with the new initial conditions,

$$\Lambda_n(\lambda_0, \lambda_1, \cdots, \lambda_{K-1}) = \begin{cases} 0, & \text{if } \lambda_1 \notin [\lambda_{1\min}, \lambda_{1\max}] \text{ or } \cdots \text{ or } \lambda_{K-1} \notin [\lambda_{(K-1)\min}, \lambda_{(K-1)\max}] \\ 1, & \text{if } \lambda_1 = \lambda_{1\max} \text{ and } \cdots \text{ and } \lambda_{K-1} = \lambda_{(K-1)\max} \\ 1, & \text{if } \lambda_1 = \lambda_{1\min} \text{ and } \cdots \text{ and } \lambda_{K-1} = \lambda_{(K-1)\min} \end{cases} \qquad (2.36)$$

where

$$\lambda_{r\min} = \sum_{j=1}^{\lambda_0} j^r \quad \text{and} \quad \lambda_{r\max} = \sum_{j=n-\lambda_0+1}^{n} j^r, \ r = 1, \cdots, K-1.$$

Note that similar initial conditions were introduced only for enumeration of dc^2-free words in [30]. Since a dc^K-free word is the complement of another dc^K-free word (two words in which "$+1$'s" and "-1's" occupy opposing positions), enumerating half of such words is enough. In this case, the recursion (2.35) starts from $N-1$ rather than N. Based on the recursion (2.35) and the new initial conditions (2.36), by using very modest computing equipment, more results of $|\mathcal{A}(N, K)|$ can be enumerated. The results obtained to date are reported in Table 2.1. A more efficient algorithm for enumerating $|\mathcal{A}(N, 2)|$ is developed in Section 2.4. The computational values of $|\mathcal{A}(N, 2)|$ are listed in Table 2.5.

Table 2.1 The number of dc^K-free words in $\mathcal{A}(N, K)$ ($N \le 64$ and $3 \le K \le 6$)

N	K			
	3	4	5	6
8	2	0	0	0
12	2	0	0	0
16	14	2	0	0
20	48	0	0	0
24	592	16	0	0
28	2886	0	0	0
32	34888	78	2	0
36	277810	0	0	0
40	3162414	4414*	2*	0
44	29524800	0	0	0
48	353955028	288938	204*	2*
52	3700663670	0	0	. 0
56		12399484	434	0*
60		0	0	0
64				6*

Note that the results with superscript "*" in the table above have also been presented in recent publications [52] and [55].

Instead of enumeration of the exact number of words in $\mathcal{A}(N,K)$, several theoretical bounds on $|\mathcal{A}(N,K)|$, or equivalently bounds on redundancy of $\mathcal{A}(N,K)$ have been developed. Denote redundancy of $\mathcal{A}(N,K)$ to be

$$r(N,K) = N - \log_2 |\mathcal{A}(N,K)|.$$

Recently, it has been shown that [29]:

$$\limsup_{N \to \infty} r(N,K) = \frac{K^2}{2} \log_2 N.\tag{2.37}$$

Roth *et al* [27] developed a lower bound on $r(N,K)$ which is given in the theorem below.

Theorem 2.6 [27]: For all $N > K \geq 1$,

$$r(N,K) \geq (K-1)(\log_2(N) - \log_2(K-1)).\tag{2.38}$$

Johnson's bounds [59, Theorems 2-5] are well-known upper bounds on the cardinality of constant-weight words with length N, minimum distance d, and word weight w, which is denoted as $A(N,d,w)$. For dc^K-free words ($K \geq 1$), from (2.15), there always exists $u^{(0)} = 0$ implying that the number of "1's" equals the number of "-1's" in the words. This is equivalent to the words over the binary alphabet $\{0,1\}$ with a constant weight $w = N/2$. Replacing $w = N/2$ into [59, Theorem 2 and Corollary 5] yields the following lemma.

Lemma 2.4: $A(N,d,N/2) \leq \left\lfloor \dfrac{2d}{2d-N} \right\rfloor$, if $d > N/2$; $\qquad\qquad$ (2.39a)

$$A(N,d,N/2) \leq \left\lfloor 2 \left\lfloor \frac{N-1}{N/2-1} \cdots \left\lfloor \frac{N/2+d/2}{d/2} \right\rfloor \cdots \right\rfloor \right\rfloor, \text{ if } d \leq N/2.\tag{2.39b}$$

In [23], it has been shown that the minimum distance d of a binary K-OZD code is at least $2K$.

Theorem 2.7: For all $N > K \geq 1$,

$$r(N,K) \geq N - \log_2 \left\lfloor \frac{4K}{4K-N} \right\rfloor, \text{ if } K > N/4;\tag{2.40a}$$

$$r(N,K) \geq N - \log_2 \left\lfloor 2 \left\lfloor \frac{N-1}{N/2-1} \cdots \left\lfloor \frac{N/2+K}{K} \right\rfloor \cdots \right\rfloor \right\rfloor, \text{ if } K \leq N/4.\tag{2.40b}$$

Table 2.2 compares Roth *et al*'s lower bound on redundancy of $\mathcal{A}(N,K)$ shown in Theorem 2.6 with the new lower bound on redundancy of $\mathcal{A}(N,K)$ presented in Theorem 2.7 and the actual values of redundancy of $\mathcal{A}(N,K)$ which are calculated from Table I in [23].

Table 2.2 Comparison of Roth *et al*'s and new lower bounds, and actual redundancy of K-OZD codes ($N \le 32, K \le 5$)

K	N											
	4			8			12			16		
	Roth's	New	Actual	Roth's	New	Actual	Roth's	New	Actual	Roth's	New	Actual
1		1.42	1.42		1.87	1.87		2.15	2.15		2.35	2.35
2	2	3	3	3	4.19	5	3.58	4.96	6.14	4	5.55	6.96
3				4	6.42	7	5.17	6.96	11	6	8.18	12.19
4										7.25	9.91	15
5												

K	N											
	20			24			28			32		
	Roth's	New	Actual	Roth's	New	Actual	Roth's	New	Actual	Roth's	New	Actual
1		2.50	2.50		2.63	2.63		2.74	2.74		2.84	2.84
2	4.32	5.96	7.59	4.58	6.33	8.10	4.81	6.65	8.54	5	6.92	8.91
3	6.64	8.66	14.42	7.17	9.14	14.79	7.61	9.88	16.51	8	10.17	16.91
4				9	11.54	20				10.25	12.76	25.71
5										12	15.22	31

2.4 Algorithms to Enumerate Codewords for Dc²-Constrained Channels

Dc²-free codes (also called dc²-balanced codes), first introduced by Immink [22], exhibit the property that both the power spectrum and its second derivative are zero at zero frequency (dc), resulting in significant suppression of spectral components at low frequencies to match dc²-constrained channels.

Let $\{x\} = \{x_1, x_2, \cdots, x_l, \cdots\}$ represent an encoded sequence, where $x_l \in \{-1, +1\}$, $l \in [1, \infty)$. As defined in (2.7), the 1st-order and 2nd-order RDS's of the sequence $\{x\}$ at instant l can be expressed as $\mathrm{RDS}_l^{(1)} = \sum_{l=1}^{n} x_l$ and $\mathrm{RDS}_l^{(2)} = \sum_{j=1}^{n} \sum_{l=1}^{j} x_l$ respectively, assuming $\mathrm{RDS}_0^{(1)} = \mathrm{RDS}_0^{(2)} = 0$.

Both $\mathrm{RDS}_l^{(1)}$ and $\mathrm{RDS}_l^{(2)}$ are bounded at an arbitrary time instant in a dc²-free code [26] and in general, a dc²-free code can be generated by concatenating codewords such that values of word-end-$\mathrm{RDS}^{(1)}$ and -$\mathrm{RDS}^{(2)}$ ($\mathrm{RDS}^{(1)}$ and $\mathrm{RDS}^{(2)}$ at the end of a word) are bounded. Let the word disparity (*WD*) and word disparity sum (*WDS*) be the $\mathrm{RDS}^{(1)}$ and $\mathrm{RDS}^{(2)}$ over the span of a codeword $\underline{x}_k = (x_{k,1}, x_{k,2}, \cdots, x_{k,N})$, where $x_{k,u} = \{-1, +1\}$, $k \in [1, \infty)$, $u = 1, 2, \cdots, N$ and $x_{k,u} = x_{(k-1)N+u}$). For convenience, let $x_{k,u} = b_{k,N+1-u}$, $u = 1, 2, \cdots, N$, i.e. $\underline{x}_k = (b_{k,N}, b_{k,N-1}, \cdots, b_{k,1})$. Then

$$WD = \sum_{u=1}^{N} x_{k,u} = \sum_{u=1}^{N} b_{k,u} \qquad (2.41)$$

$$WDS = \sum_{j=1}^{N} \sum_{u=1}^{j} x_{k,u} = \sum_{u=1}^{N} u \cdot b_{k,u} \ . \tag{2.42}$$

Evaluation of code capacity and construction of dc^2-free codes requires knowledge of the number of available codewords with specified values of WD and WDS. A technique for determining the number of such codewords has been developed [22]. Initial conditions for this algorithm have been reconsidered to reduce the computational complexity [30], however, to realize highly efficient dc^2-free codes, a long codeword length is needed [30], implying that alternative efficient enumeration techniques should be sought.

In this section, two new algorithms for evaluating the number of codewords in dc^2-free codes are developed. A new method for enumerating codewords with specified values of WD and WDS is first introduced, and then the challenge of evaluating the number of dc^2-free codewords is approached using techniques developed in the study of partitions in combinatorics. Two new algorithms based on this method of enumeration are outlined. Computation results that compare the new approaches and a recently published algorithm [30] are presented, demonstrating the numerical efficiency of the new algorithms.

2.4.1 Enumeration of Codewords

A. Enumeration process

Let an encoded sequence $\{x\}$ be formed through concatenation of codewords of length N, and let the kth codeword $\underline{x}_k$ be as defined above. Let B^+ and B^- be the number of "$+1$'s" and the number of "-1's" in a codeword, respectively. Also let $I = \{i_{B^+}, i_{B^+-1}, \cdots, i_1\}$ and $J = \{j_{B^-}, j_{B^--1}, \cdots, j_1\}$ be the sets of indices of u when $b_{k,u} = x_{k,N+1-u} = +1$ and $b_{k,u} = x_{k,N+1-u} = -1$, respectively, where $N \geq i_{B^+} > i_{B^+-1} > \cdots > i_1 \geq 1$ and $N \geq j_{B^-} > j_{B^--1} > \cdots > j_1 \geq 1$.

Characteristic 2.1: It is straightforward to show that a desired value of word disparity, WD_{des}, is given by $WD_{des} = B^+ - B^-$, and that $B^+ = (N + WD_{des})/2$ and $B^- = (N - WD_{des})/2$. When the leading B^+ positions are "$+1$'s", the value of WDS is maximal; when the leading B^- positions are "-1's", the value of WDS is minimal. Let $WDS_{max}^{(WD_{des})}$ and $WDS_{min}^{(WD_{des})}$ denote the maximum and minimum values of WDS for WD_{des}. From (2.42) there is:

$$WDS_{max}^{(WD_{des})} = \sum_{u=B^-+1}^{N} u - \sum_{u=1}^{B^-} u = (1/2)N(N+1) - B^-(B^-+1). \tag{2.43}$$

28

Similarly

$$WDS_{\min}^{(WD_{des})} = -(1/2)N(N+1) + B^+(B^+ +1).\tag{2.44}$$

Characteristic 2.2: Exchanging the positions of two symbols with different values in a word $\underline{x}_k$ will not change the value of *WD*, but it will alter *WDS*. Every exchange between two different symbols in which the symbol "+1" is moved closer to the start of the word results in an increase in *WDS*. This increment is twice the value of the difference of positions between the two symbols, which is easily verified by considering the difference in *WDS* between two words in which the symbols in positions u_q and u_p ($u_q > u_p$), are exchanged. Using (2.42), it can be shown that $\Delta wds = (u_q - u_p)(b_{k,u_p} - b_{k,u_q}) = 2(u_q - u_p)$. Note that before the exchange, $b_{k,u_p} = +1$ and $b_{k,u_q} = -1$. Denote the value of the difference $d_p = u_q - u_p$ to be the *distance* of these two symbols, and the increment of *WDS*, which is twice this value, to be Δwds.

These two characteristics are employed to enumerate all possible words with a desired value of word disparity, WD_{des}. The uniqueness and completeness of this enumeration process are demonstrated when it is interpreted in terms of restricted partitions in Section 2.4.1 B.

Enumeration rules:

1) Given WD_{des}, if the desired value of *WDS*, WDS_{des}, equals $WDS_{\min}^{(WD_{des})}$, only one word is available. This word is called the *original word*. If WDS_{des} is larger than $WDS_{\min}^{(WD_{des})}$, enumeration begins from the original word and proceeds according to the following rules.

2) Every "+1" can be exchanged only once when forming a valid word from the original word and these symbols are exchanged consecutively starting from the symbol "+1" that occupies the most significant position within the "+1's" in the original word. The maximum number of exchanges is B^+. After each exchange, a new word which is called the *current word* is generated. The next exchange starts from this current word.

3) Each exchange must be made between two different symbols, e.g., the pth "+1" and qth "−1" in the original or a current word. It is required that $i_p < j_q$, where $i_p \in I$, $j_q \in J$. The distance d_p equals $j_q - i_p$.

4) When the p_1th and p_2th "+1's" ($B^+ \geq p_2 > p_1 \geq 1$) are involved in exchanges, d_{p_2} must be greater than or equal to d_{p_1}. Therefore in every word an exchange must be made where $B^- \geq d_p \geq 1$.

5) During the formation of a valid word, the increment in WDS from the original word, which is defined to be $2S$, is the sum of individual increments of Δwds_p ($\Delta wds_p = 2d_p$) which are obtained when the pth "+1" is involved in an exchange.

6) If $2S = WDS_{des} - WDS_{min}^{(WD_{des})}$, a valid word with the specified word length and the desired values of WD and WDS has been generated.

Example 2.2: Enumeration of codewords with $N = 4$ and $WD_{des} = 0$:

In this example all codewords of length $N(=4)$ with zero disparity are enumerated. Since these values have zero disparity, they contain an equal number of "+1's" and "-1's". Let $A_{N,WD_{des}}$ be the total number of codewords for a given N and WD_{des}. Taking into account all possible compositions of $B^+(=2)$ "+1's" and $B^-(=2)$ "-1's" in $N(=4)$ positions yields $A_{4,0} = C_2^4 = 6$.

Let the subscripts on the symbols "+1" or "-1" represent the positions of these symbols in the original word, and let the double arrow $\leftrightarrow$ denote an exchange between the specified symbols. The rules given above result in the enumeration process shown in Figure 2.7.

$$-1_4\ -1_3\ +1_2\ +1_1\ (WD_{des} = 0,\ WDS = -4)$$

$+1_2 \leftrightarrow -1_3$ ①   ② $+1_2 \leftrightarrow -1_4$

$-1_4\ +1_2\ -1_3\ +1_1\ (0, -2)$ $+1_2\ -1_3\ -1_4\ +1_1\ (0, 0)$
$2S = \Delta wds_2 = 2$ $2S = \Delta wds_2 = 4$

$+1_1 \leftrightarrow -1_3$ ③ $+1_1 \leftrightarrow -1_4$ ④   ⑤ $+1_1 \leftrightarrow -1_3$

$-1_4\ +1_2\ +1_1\ -1_3\ (0, 0)$ $+1_2\ -1_3\ +1_1\ -1_4\ (0, 2)$ $+1_2\ +1_1\ -1_4\ -1_3\ (0, 4)$
$2S = \Delta wds_2 + \Delta wds_1$ $2S = \Delta wds_2 + \Delta wds_1$ $2S = \Delta wds_2 + \Delta wds_1$
     $= 2+2$ $= 4+2$ $= 4+4$
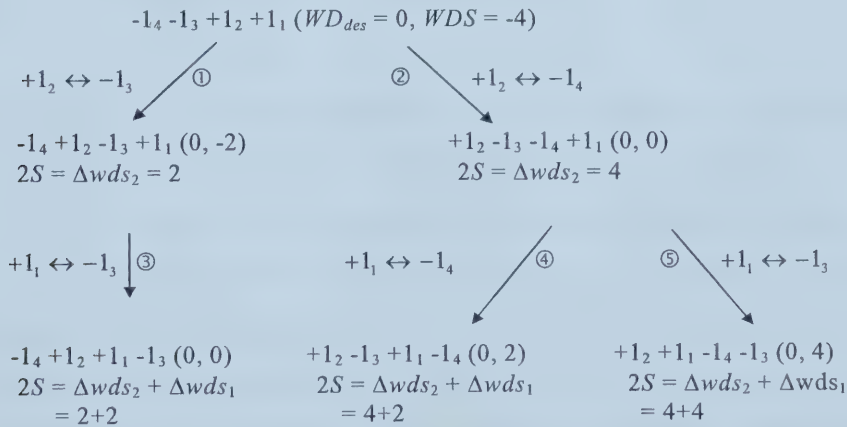
Figure 2.7 Enumeration of codewords with $N = 4$ and $WD_{des} = 0$.

Branches ① and ② result from exchange of $+1_2$, the most significant "+1" within the "+1's" in the original word, with -1_3 and -1_4, respectively. Exchanging $+1_2$ with -1_3 on branch ① results in the word $-1_4 +1_2 -1_3 +1_1$ where WDS is $2S$ (= 2) greater than the original word. There is only one further exchange from this current word that satisfies the enumeration rules, that being the exchange of $+1_1$ with -1_3, resulting in a word with WDS (= 2+2 = 4)

greater than the original word. Following the exchange on branch ②, two different exchanges are possible, as indicated by branches ④ and ⑤. After these exchanges, no further movement satisfies our enumeration rules, and the enumeration is complete.   ◊

In addition, there is an alternative enumeration method in which the original word is assumed to be the codeword with $WDS_{\max}^{(WD_{des})}$ and exchanges are performed between a symbol "+1" (ahead) and a symbol "−1" (behind). Similar enumeration rules ensure this kind of enumeration to be unique and complete. Given a complementary original word pair (complementary words are defined in Section 2.3.3) which have same absolute values of WD and WDS, but different signs, the two enumeration methods will generate a complementary codeword pair through the same sequence of exchanges.

B. *Interpretation with restricted partitions*

The enumeration technique described above can be associated with restricted partitions [60, ch.6]. A partition of a natural number n into k parts with restrictions on the number of parts and the maximum value of parts is a representation of n in the form:

$$n = d_k + d_{k-1} + \cdots + d_1,$$ (2.45)

where the natural numbers $d_k, d_{k-1}, \cdots, d_1$ (the parts) satisfy

$$m \geq d_k \geq d_{k-1} \geq \ldots \geq d_1 \geq 1 \ (m \geq 1).$$ (2.46)

To associate the new enumeration technique with restricted partitions, let the parameters in (2.45) and (2.46) be $n = S = (WDS_{des} - WDS_{\min}^{(WD_{des})})/2$, $B^+ \geq k \geq 1$, $m = B^-$, and the numbers $d_k, d_{k-1}, \cdots, d_1$ be values of d_p as described in the enumeration rules. As discussed in [60, ch.6], owing to condition (2.46) the order of integers $d_k, d_{k-1}, \cdots, d_1$ is fixed, so a partition is unique when it is a specified collection of integers. Also, if all possible partitions of the natural number n into k parts with restrictions can be enumerated, the partition is complete.

The uniqueness and completeness of the enumeration technique outlined above are demonstrated below. The enumeration rules above ensure a one-to-one mapping between the formation of a valid codeword and a restricted partition. Enumeration rules 2) and 3) ensure that only k consecutive "+1's" in the most significant positions within the "+1's" in the original word are involved in exchanges so as to avoid the generation of duplicate partitions. Rule 4) ensures that the order of Δwds_p satisfies the condition (2.46), therefore there is a one-to-one mapping between an enumeration and a restricted partition. Enumeration rules 3) and 4) also ensure that the collection of Δwds_p is unique, and uniqueness of the enumeration follows.

Regarding completeness, it is noted that partitions $n = d_k + d_{k-1} + \cdots + d_1$, where $m \geq d_k \geq d_{k-1} \geq \ldots \geq d_1 \geq 1$ ($m \geq 1$), can be representations of all possible k collected integers satisfying the given condition. Also due to the one-to-one mapping between an enumeration of a word and a restricted partition described above, all possible codewords with specified length N, WD_{des} and $WDS_{min}^{(WD_{des})} \leq WDS_{des} \leq WDS_{max}^{(WD_{des})}$ can be enumerated.

Example 2.3: Enumeration of codewords with length $N = 4$ and $WD_{des} = WDS_{des} = 0$.

For $N = 4$, $WDS_{min}^{(0)} = -4$, $n = (WDS_{des} - WDS_{min}^{(WD_{des})})/2 = 2$ and $B^+ = B^- = 2$. Using these values of n, B^+, and B^- in (2.45) and (2.46) two available partitions are found: 2=2 and 2=1+1. Enumeration of codewords in accordance with these partitions is shown in Figure 2.8.

$$-1_4 -1_3 +1_2 +1_1 \ (WD_{des} = 0, \ WDS_{min}^{(0)} = -4)$$

$+1_2 \leftrightarrow -1_3$ ①   ② $+1_2 \leftrightarrow -1_4$

$-1_4 +1_2 -1_3 +1_1 \ (0, -2)$ $+1_2 -1_3 -1_4 +1_1 \ (0, 0)$
                                 partition: $n = d_2 = 2$

$+1_1 \leftrightarrow -1_3$ ③

$-1_4 +1_2 +1_1 -1_3 \ (0, 0)$
partition: $n = d_2 + d_1 = 1+1$
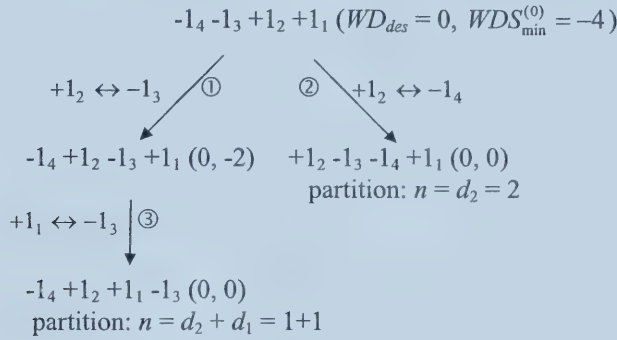
Figure 2.8 Enumeration of codewords with $N = 4$ and $WD_{des} = WDS_{des} = 0$.

In Figure 2.8, both exchanges ① and ② begin from the original word and involve exchange of $+1_2$. Branch ② results in a word with the required *WDS*, but branch ① requires another exchange to generate the other codeword with this *WDS*. This simple example demonstrates that restricted partitions can be used when enumerating codewords. ◊

In Section 2.4.2, the above result is employed to derive two efficient algorithms for calculating the cardinality of codewords with specified *WD* and *WDS*.

2.4.2 Algorithms

A. Algorithm I

Algorithm I is an efficient method of enumerating codewords with specified values of *WD* and *WDS*. As described above, this type of enumeration is associated with the problem of restricted partitions. Computational results in Section 2.4.3 show that this algorithm involves

fewer computations than the method of enumerating these codewords recently published in [30], which is the most efficient enumeration method published to date.

Generating functions are useful for enumerating partitions [60, ch.2]. By combining the generating function for the partition with restriction on part values [60, p.111 (2)] and the generating function for the partition with exactly fixed parts [60, p.112 (6)], a two-variable generating function to enumerate partitions of the natural number n into k parts in which parts can be repeated but in which no parts can have a value greater than m is formed. This generating function is:

$$F_m(t,u) = 1/(1-tu)(1-t^2u) \cdots (1-t^m u) = \sum_{n,k} f_m(n,k) t^n u^k , \qquad (2.47)$$

where the coefficient $f_m(n,k)$ represents the number of restricted partitions. From (2.47), there is

$$F_m(t,u) = F_{m-1}(t,u)/(1-t^m u)$$

and

$$\sum_{n,k} f_m(n,k) t^n u^k = \left(1/(1-t^m u)\right) \sum_{n,k} f_{m-1}(n,k) t^n u^k .$$

Therefore there exists the recursive relation:

$$f_m(n,k) = f_{m-1}(n,k) + f_m(n-m,k-1) \qquad (2.48)$$

with the initial conditions

$$f_m(n,k) = \begin{cases} 0, & \text{if } n < k \text{ or } n > m \cdot k \\ 1, & \text{if } k = 1 \text{ and } m \geq n \\ 1, & \text{if } n = k \text{ or } n = m \cdot k \\ 1, & \text{if } n = k+1 \text{ or } n = m \cdot k - 1 \\ f_{l+1}(2l,l), & \text{if } m \geq l+1, k \geq l \text{ and } n = k+l \text{ (or } n = m \cdot k - l), \quad l = 2, 3, \cdots \end{cases} \qquad (2.49)$$

The values of $f_{l+1}(2l,l)$ for relatively small l (up to 10) are given in Table 2.3. Using this result, the following expression for enumerating the total number of codewords with a given length N and desired WD and WDS is given. Denote this value to be $A_{N,WD_{des},WDS_{des}}$:

$$A_{N,WD_{des},WDS_{des}} = \begin{cases} 1, & \text{if } WDS_{des} = WDS_{max}^{(WD_{des})} \text{ or } WDS_{des} = WDS_{min}^{(WD_{des})} \\ f_m(n,1) + f_m(n,2) + \cdots + f_m(n,K), & \text{otherwise} \end{cases} , \qquad (2.50)$$

where n and m are as defined above, and $K = B^+$. Let the set of words with length N and with both zero WD and WDS values be $\mathcal{A}(N,2)$. An application of Algorithm I is to efficiently enumerate the number of codewords in $\mathcal{A}(N,2)$. For calculation of $A_{N,0,0}$, because a word in

$\mathcal{A}(N, 2)$ that is the complement of another word in $\mathcal{A}(N, 2)$ can always be found, enumerating half of such words in $\mathcal{A}(N, 2)$ is sufficient. In this case, let $m = (N/2)-1$.

Table 2.3 List of $f_{l+1}(2l, l)$, $l = 2, 3, \cdots, 10$.

l	$f_{l+1}(2l, l)$
2	2
3	3
4	5
5	7
6	11
7	15
8	22
9	30
10	42

Example 2.4: Calculation of $A_{8,0,0}$ by Algorithm I.

Consider calculation of the cardinality of codewords with $N = 8$ and $WD_{des} = WDS_{des} = 0$, i.e. $|\mathcal{A}(8, 2)|$. The parameters in (2.50) can be determined by $m = N/2-1 = 3$, $K = 4$, and $n = 8$. In this case, $|\mathcal{A}(8, 2)| = 2[f_3(8, 1) + f_3(8, 2) + f_3(8, 3) + f_3(8, 4)]$. Combining (2.48) and (2.49) yields:

$$|\mathcal{A}(8, 2)| = 2[\underbrace{f_3(8, 1)}_{=0} + \underbrace{f_3(8, 2)}_{=0} + f_3(8, 3) + f_3(8, 4)]$$

where

$$f_3(8, 3) = \underbrace{f_2(8, 3)}_{=0} + f_3(5, 2) = \underbrace{f_2(5, 2)}_{=0} + \underbrace{f_3(2, 1)}_{=1} = 1$$

and

$$f_3(8, 4) = \underbrace{f_2(8, 4)}_{=1} + f_3(5, 3) = 1 + f_2(5, 3) + \underbrace{f_3(2, 2)}_{=1}$$
$$= 1 + \underbrace{f_1(5, 3)}_{=0} + f_2(3, 2) + 1$$
$$= 1 + \underbrace{f_1(3, 2)}_{=0} + \underbrace{f_2(1, 1)}_{=1} + 1 = 3.$$

So $|\mathcal{A}(8, 2)| = 8$, which can be verified by listing all binary words in $\mathcal{A}(8, 2)$ as in Table 2.4 below.

Table 2.4 List of available codewords with $N = 8$ and $WD_{des} = WDS_{des} = 0$

No.	Codeword	WD_{des}	WDS_{des}
1	00111100	0	0
2	01011010	0	0
3	01100110	0	0
4	01101001	0	0
5	10010110	0	0
6	10011001	0	0
7	10100101	0	0
8	11000011	0	0

◊

By employing Algorithm I, using very modest computing equipment, $A_{N,0,0}$ for $N \le 52$ as well as the maximal possible source word length can be evaluated. Recall that, as described in Section 2.3.2, N must be a multiple of four for dc^2-free words. These results are tabulated in Table 2.5. Note that the values of $A_{N,0,0}$ (denoted $|\mathcal{A}(N,2)|$) can also be approximated based on the work in [29], [30]. Table 2.5 also lists the approximate values of $A_{N,0,0}$, $|\tilde{\mathcal{A}}(N,2)|$, as well as the percentage difference between $|\mathcal{A}(N,2)|$ and $|\tilde{\mathcal{A}}(N,2)|$. Clearly, when N is relatively large, the approximation approaches the value of $A_{N,0,0}$. The complexity of Algorithm I and the algorithm developed in [30] is compared in Section 2.4.3.

Table 2.5 Exact and approximate cardinality of $\mathcal{A}(N,2)$, percentage difference between the cardinality values, and maximal source word length

N	$|\mathcal{A}(N,2)|$	$|\tilde{\mathcal{A}}(N,2)|$	Percentage difference	$\lfloor \log_2 |\mathcal{A}(N,2)| \rfloor$
4	2	2	0	1
8	8	8	0	3
12	58	62	6.90	5
16	526	564	7.22	9
20	5448	5781	6.11	12
24	61108	64234	5.12	15
28	723354	755082	4.39	19
32	8908546	9249763	3.83	23
36	113093022	116935287	3.40	26
40	1470597342	1515481327	3.05	30
44	19499227828	20039422508	2.77	34
48	262754984020	269418902609	2.54	37
52	3589093760726	3673024565756	2.34	41

B. Algorithm II

To improve the code rate, dc^2-free codes can also be generated using state-dependent encoding [22], [26]. For these codes, it is of interest to enumerate codewords with a specified value of WD and a value of WDS that falls within a specified range. The Tallini-Bose algorithm [30] and Algorithm I presented above can be successively employed for this purpose, however, in this subsection, another algorithm better suited to this situation is introduced. Specifically the case when the value of WD is given is considered. It is desirable to evaluate the number of available codewords whose value of WDS is less than or equal to an upper value of WDS, WDS_u. To do so, an *extended restricted partition* model is employed.

Extended restricted partitions are partitions of the natural number n ($n = 1, 2, \cdots, S$) into k ($k = 1, 2, \cdots, K$) parts with restrictions on the maximum value of parts, m. The difference with extended restricted partitions is that the values of n and k are not fixed as in restricted partitions, but fall within a range.

If $S \geq m \cdot K$, partitions are defined to be *complete extended restricted partitions* because they include all possible partitions satisfying $m \cdot K \geq n \geq 1$, otherwise they are defined to be *incomplete extended restricted partitions*. Let $cp(m, K)$ and $icp(m, K, S)$ be the number of solutions of complete and incomplete extended restricted partitions, respectively, where $m, K, S \geq 1$. In the application of enumerating the number of codewords with given WD_{des} and WDS_{des} that falls in a range, $m = B^-$, $K = B^+$ and $S = (WDS_u - WDS_{\min}^{(WD_{des})})/2$, where $WDS_u \leq WDS_{\max}^{(WD_{des})}$. Alternatively, similar to the descriptions in Section 2.4.2 A, if it is desired to evaluate the number of available codewords whose value of WDS is larger than or equal to a lower bound for WDS, WDS_l, define $m = B^+$, $K = B^-$ and $S = (WDS_{\max}^{(WD_{des})} - WDS_l)/2$, where $WDS_l \geq WDS_{\min}^{(WD_{des})}$.

1) Complete extended restricted partitions

Let m, K, S, and N be defined as above. In the application regarding the enumeration of words, $S = (WDS_{\max}^{(WD_{des})} - WDS_{\min}^{(WD_{des})})/2 = m \cdot K$. By means of the model of complete extended restricted partitions, all possible words except for the original word with K "$+1$'s" and m "-1's" (because $m \cdot K \geq n \geq 1$) are enumerated. It is known that the number of words with K "$+1$'s" and m "-1's" is a combination of C_K^{m+K} (note that $N = m + K$). So there is:

$$cp(m, K) = C_K^N - 1. \tag{2.51}$$

Example 2.5: Enumeration of complete extended restricted partitions with $m = 3$ and $K = 2$.

From (2.51), it can be found that $cp(3, 2) = C_2^5 - 1 = 9$ which agrees with complete extended restricted partitions with $m = 3$ and $K = 2$ which are listed below.

$$1 = 1; \ 2 = 1 + 1 = 2; \ 3 = 2 + 1 = 3; \ 4 = 2 + 2 = 3 + 1; \ 5 = 3 + 2; \ 6 = 3 + 3. \qquad \Diamond$$

2) Incomplete extended restricted partitions

Let m, K, and S be defined as above. A recursive method to calculate $icp(m, K, S)$ is developed as follows:

$$icp(m, K, S) = \begin{cases} cp(m, K), & \text{if } S \geq m \cdot K \\ S, & \text{if } m = 1, K \geq S \\ K, & \text{if } m = 1, K < S \\ S, & \text{if } K = 1, m \geq S \\ m, & \text{if } K = 1, m < S \\ 1 + icp(m - 1, K, S) + icp(m, K - 1, S - m), & \text{if } m, K > 1, S \geq m \\ icp(m - 1, K, S), & \text{if } m, K > 1, S < m \\ 0, & \text{otherwise} \end{cases} \qquad . \text{ (2.52)}$$

Example 2.6: Evaluation of $icp(3, 3, 6)$. It is straightforward to illustrate the partitions $icp(3, 3, 6)$ in a Ferrers graph [60, ch.6]; see Figure 2.9. From Figure 2.9, it is straightforward to find that $icp(3, 3, 6) = 15$. This same result is obtained using the recursions in (2.52); see Example 2.7.
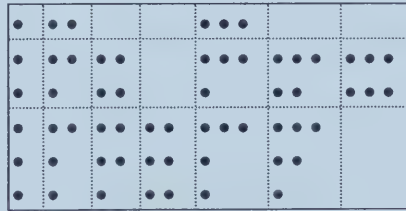
Figure 2.9 Illustration of the partitions $icp(3, 3, 6)$.

Consider application of the above results to the evaluation of the number of codewords. Let $A_{N, WD_{des}}$ be the total number of codewords with length N and desired WD. It is straightforward to show that:

$$A_{N, WD_{des}} = \binom{N}{(N + WD_{des})/2} = \binom{N}{(N - WD_{des})/2} = 1 + cp(m, K).$$

Let $A_{N,WD_{des}}(WDS_u)$ be the number of available codewords with N, WD_{des} and values of word disparity sum in the range $\left[WDS_{min}^{(WD_{des})}, WDS_u\right]$. Let $S = (WDS_u - WDS_{min}^{(WD_{des})})/2$. Then:

$$A_{N,WD_{des}}(WDS_u) = \begin{cases} 1, & \text{if } WDS_u = WDS_{min}^{(WD_{des})} \\ A_{N,WD_{des}}, & \text{if } WDS_u = WDS_{max}^{(WD_{des})} \\ 1 + icp(B^-, B^+, S), & \text{otherwise} \end{cases}.$$

Let $A_{N,WD_{des}}(WDS_1, WDS_2)$ be the number of available codewords with N, WD_{des} and values of word disparity sum in the range $[WDS_1, WDS_2]$, where $WDS_2 \geq WDS_1$. Then:

$$A_{N,WD_{des}}(WDS_1, WDS_2) = \begin{cases} A_{N,WD_{des}}(WDS_2), & \text{if } WDS_1 = WDS_{min}^{(WD_{des})} \\ A_{N,WD_{des}}(WDS_2) - A_{N,WD_{des}}(WDS_1 - 2), & \text{if } WDS_1 > WDS_{min}^{(WD_{des})} \end{cases}.$$

An example calculation of $A_{N,WD_{des}}(WDS_1, WDS_2)$ is given below.

Example 2.7: Calculation of $A_{N,WD_{des}}(WDS_1, WDS_2)$ by Algorithm II.

Consider calculation of the cardinality of codewords with $N = 6$, $WD_{des} = 0$, and WDS that falls within the range $[-3, 3]$, i.e. $A_{6,0}(-3, 3)$, which equals $A_{6,0}(3) - A_{6,0}(-5)$. Then $m = B^- = 3$, $K = B^+ = 3$, and $WDS_{min}^{(0)} = -9$. The calculation proceeds as follows:

1) $WDS_u = 3$ (i.e. $S = 6$):

$$\begin{aligned} icp(3, 3, 6) &= 1 + icp(2, 3, 6) + icp(3, 2, 3) \\ &= 1 + cp(2, 3) + 1 + icp(2, 2, 3) + icp(3, 1, 0) \\ &= 1 + cp(2, 3) + 1 + 1 + icp(1, 2, 3) + icp(2, 1, 1) \\ &= 1 + 10 - 1 + 1 + 1 + 2 + 1 \\ &= 15. \end{aligned}$$

2) $WDS_u = -5$ (i.e. $S = 2$):

$$\begin{aligned} icp(3, 3, 2) &= icp(2, 3, 2) \\ &= 1 + icp(1, 3, 2) + icp(2, 2, 0) \\ &= 3. \end{aligned}$$

Note that the value of $icp(3, 3, 2)$ can also easily be verified from Figure 2.9.

3) $A_{6,0}(-3, 3) = A_{6,0}(3) - A_{6,0}(-5) = icp(3, 3, 6) - icp(3, 3, 2) = 12$.

To verify this result, in Table 2.6 all possible codewords that satisfy these conditions are listed.

Table 2.6 List of available codewords with $N = 6$, $WD_{des} = 0$ and $WDS_{des} \in [-3, 3]$

No.	Codeword	WD_{des}	WDS_{des}
1	001110	0	-3
2	010101	0	-3
3	010110	0	-1
4	011001	0	-1
5	011010	0	1
6	011100	0	3
7	100011	0	-3
8	100101	0	-1
9	100110	0	1
10	101001	0	1
11	101010	0	3
12	110001	0	3

2.4.3 Comparison of Efficiency

There are many ways to measure and compare the efficiency of algorithms. Here, only time complexity is considered.

To enumerate the number of desired words, the Tallini-Bose algorithm [30] and both new algorithms introduced in this section employ recursion relations with appropriate initial conditions. Since recursion is an innate characteristic of tree structures [57, ch.2.3], and trees can be described in terms of their nodes, the number of nodes visited by each algorithm is used to compare the efficiency of the algorithms. At each node, recursion may either continue or halt depending on the given initial conditions, and each node could be visited several times. Since each visited node is associated with arithmetic operations (addition, subtraction and/or comparison) which affect computer running times, the total number of visited nodes is counted to compare the three algorithms for various cases.

The efficiencies of the Tallini-Bose algorithm and Algorithm I (with $l = 5$ in initial conditions (2.49)) are compared in Figure 2.10 for enumeration of the cardinality of $\mathcal{A}(N, 2)$. This figure shows that the number of visited nodes for Algorithm I is about one order of magnitude lower than the number of nodes visited in the Tallini-Bose algorithm when N is relatively large ($4 \leq N \leq 44$, N a multiple of 4). Note that the larger the value of l, the fewer the number of visited nodes of Algorithm I, however, more computational operations are required to access some nodes. Choosing an optimum value of l for general cases is unclear. For N up to 44, it has been found that $l = 5$ offers a good tradeoff, and its use yields CPU running times of about 7 times less for Algorithm I than for the Tallini-Bose algorithm.
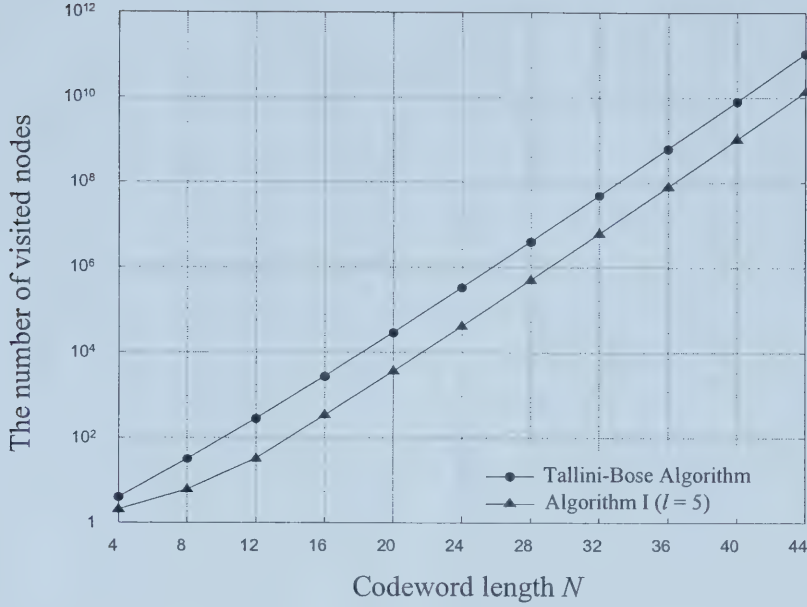
Figure 2.10 Comparison of the number of visited nodes for the Tallini-Bose algorithm and Algorithm I ($l = 5$).

Figure 2.11 compares the number of visited nodes for Algorithm I ($l = 5$) and Algorithm II when $N = 24$ and 32, $WD = 0$ and $WDS = [-B, B]$ for $2 \leq B \leq 128$. Denote B to be the WDS boundary. Figure 2.11 demonstrates that the number of visited nodes in Algorithm II is slightly larger than Algorithm I when B is small, and that the number of visited nodes in Algorithm II is much smaller than Algorithm I when B is large. The tests show that, for the case of $N = 32$, $WD = 0$ and $WDS = [-128, 128]$, CPU running times for Algorithm II are over 100 times less than those for Algorithm I. This highlights the advantage of Algorithm II over Algorithm I when the word length and word disparity values are fixed, and the word disparity sum values are allowed to fall within a large range.
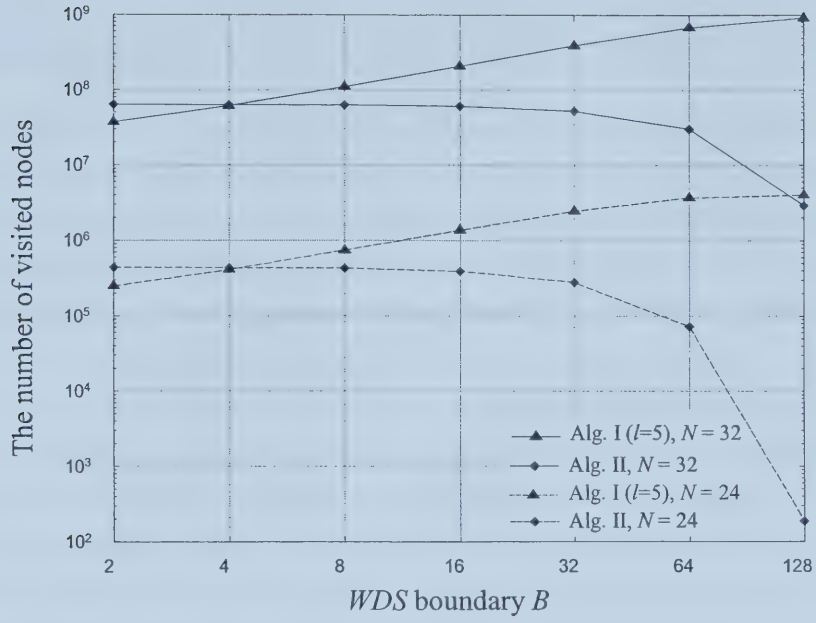
Figure 2.11 Comparison of Algorithms I ($l = 5$) and II for $N = 24$ and 32, $WD = 0$ and $WDS = [-B, B]$.

Chapter 3
PERFORMANCE EVALUATION OF DC^K-FREE CODES [4]

As described in Section 2.1, the power spectra of channel-input signals in synchronous storage systems are generally shaped through controlling the power spectra of the discrete symbol sequences of the signals by coding techniques rather than by controlling the power spectrum of the pulse, which is a rectangular waveform. In this chapter, evaluation of the spectral suppression performance at low frequencies resulting from dc^K-free coding is considered.

In Section 2.2.2, a symbol-by-symbol dc^K-free encoder was modeled as a finite-state sequential machine and illustrated by a finite-state transition diagram (dc^K-FSTD), where the coded symbol sequence is the output of the FSSM. To limit error propagation during decoding, practical constrained codes employ block coding techniques [44], [47]. In this chapter, block codes are categorized as fixed-length or variable-length, and state-independent or state-dependent. Calculation of the power spectra of block coded sequences generated through concatenation of fixed-length and variable-length words is discussed in Section 3.1. Also power spectra of high-order zero-disparity codes are calculated by extending an enumeration method for second-order zero-disparity codes [22] with new initial conditions.

The well-known performance metric for first-order spectral-null codes, sum variance, is reviewed in Section 3.2, and in Section 3.3, a new performance metric for arbitrarily high-order spectral-null codes, low-frequency spectrum-weight (LFSW), is developed and evaluated for various cases of fixed-length HOSN codes. Sum variance evaluates the width of spectral nulls, whereas LFSW evaluates the depth of spectral nulls.

Sum variance and low-frequency spectrum-weight are the only two metrics developed to date which are indicative of evaluating the performance of codes with a spectral null at dc with time-domain statistic parameters of the coded sequences. Evaluation of low-frequency spectrum weight of state-independent variable-length HOSN codes will be discussed in Chapter 5.

3.1 Power Spectra of Block Coded Sequences

3.1.1 Dc^K-Free Block Codes

In block coding, a length-λ symbol sequence defined to be a codeword with length λ is mapped to a length-μ frame of source symbols according to the recognized coding rule; both λ and μ are integers. Similar to the definition of FSSM of the symbol-by-symbol encoder

---

[4] A version of this chapter has been submitted in part for publication. Y. Xin and I. J. Fair. *IEEE Trans. Inform. Theory.*

described in Section 2.2.2, a block encoder can be modeled by a FSSM [34] that is represented by the quintuple $\mathcal{E} = (\mathbb{Y}^\mu, \mathbb{X}^\lambda, \mathcal{S}, g, h)$ where $\mathbb{Y}^\mu \subseteq \mathcal{I}^\mu$ is the set of input length-μ words $\underline{y}_n = [y_{n,1}, y_{n,2}, \cdots, y_{n,\mu}]$; $\mathbb{X}^\lambda \subseteq \mathcal{I}^\lambda$ is the set of output length-λ words $\underline{x}_n = [x_{n,1}, x_{n,2}, \cdots, x_{n,\lambda}]$ ($\underline{y}_n$ and $\underline{x}_n$ are the source word and codeword during encoding interval n); $\mathcal{S} = \{\sigma_1, \cdots, \sigma_S\}$ is the set of encoder states and S is the number of states; g is the state transition function, and h is the output function. Let the state at time instant n be denoted by s_n. Therefore $s_{n+1} = g(s_n, \underline{y}_n)$. The output word of the FSSM at time instant n is $\underline{x}_n = h(s_n)$ for a Moore machine or equivalently, $\underline{x}_n = h(s_n, \underline{y}_n)$ for a Mealy machine. In general, to implement state-independent decoding, the function h is of the form that during decoding the reverse function h^{-1} uniquely maps a received codeword to a source word without knowledge of encoding states.

If the lengths of source words and codewords in a code are fixed, the code is called a *fixed-length* code. For fixed-length codes, as defined in Section 2.1.1, the source word length and codeword length are denoted as M and N respectively. Alternatively, if the source words length μ and the codeword length λ of a code vary, the code is called a *variable-length* code. The variable-length code that has found greatest application is the *synchronous variable-length* code in which the ratio of λ and μ is a constant and λ is a multiple of a basic word length [1, ch.7.2], [44]. Variable-length codes that employ words with short lengths more frequently than words with longer lengths may reduce encoder and decoder complexity compared to fixed-length codes of similar efficiency and sequence properties [44]. Several practical variable-length RLL codes have been reviewed in [1, ch.7.2]. A typical variable-length dc[1]-free code, the VL43 code, was proposed in [44]. To date, however, variable-length high-order spectral-null codes have not been reported in the literature.

A dc^K-free fixed-length encoder can also be represented by a FSTD G^N which is the Nth extension of the corresponding dc^K-FSTD G given the constraints [26]. The properties of G^N have been given in [26] and include: 1) the nodes (states in the corresponding FSSM) in G^N are the nodes in G which are represented by RDS$^{(r)}$ ($r = 1, \cdots, K$) values[5]; 2) for each pair of states $(s_l = \sigma_i, s_{l+\lambda} = \sigma_j)$ in G, there exists a directed edge in G^N from σ_i to σ_j labeled with the corresponding length-N symbol sequence which is the output corresponding to the path from σ_i to σ_j in G.

---

[5] throughout this thesis, RDS$^{(r)}$ ($r = 1, \cdots, K$) values in G^N are considered only at the end of words

A block encoder can also be categorized as either a state-independent encoder or a state-dependent encoder. In a state-independent dc^K-free encoder, there exist unique one-to-one mappings between source words and codewords. Each codeword is a dc^K-free word that is independent of encoding state information. Kth-order zero-disparity codes discussed in Section 2.3 are straightforward state-independent dc^K-free codes that are generated through free concatenation of dc^K-free words with the same word length [23]. To increase the efficiency of K-OZD codes, the word length has to be increased [29], resulting in an increase in the complexity of the encoder and decoder as well as increased error propagation during decoding.

Definition 3.1 [44]: For a fixed-length encoder with binary input, principal states in the FSTD are those states from which at least 2^M edges branch off where each of the edges connects with another principal state. The necessary and sufficient condition for the existence of a constrained code is the existence of the set of principal states.

For a state-independent block code, there is only one principal state [26]. Figure 3.1 shows a simple example of FSTD G and the 4th-extension of G for the 1/4 state-independent dc^2-free code (1/4 2-OZD code). In this case, both codewords "$+--+$" and "$-++-$" are dc^2-free words that are uniquely mapped to the binary source word "1" and "0" respectively. The large circle in Figure 3.1 (b) represents the principal state.

(a)   (b)
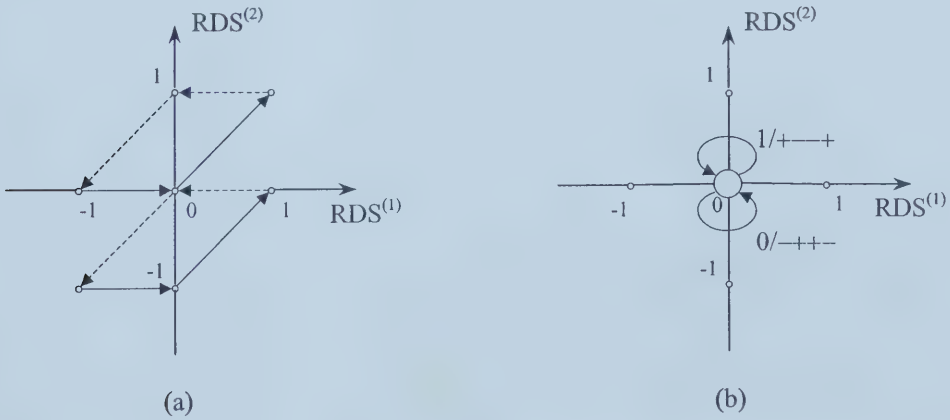
Figure 3.1   (a) FSTD of the 1/4 2-OZD code G (solid lines and dashed lines indicated by +1 and −1 respectively); (b) 4th-extension of the FSTD for the 1/4 2-OZD code G^4.

Alternatively, during state-dependent encoding, in the Mealy machine FSM model the output word and the next state depend on not only the source word but also the current state. The state sequence is a Markov chain and the output codewords are correlated rather than mutually independent. It has been noted [22], [25] that in general, for a given word length, state-

dependent block dc^K-free encoding can realize a higher code rate than state-independent block dc^K-free encoding. An intuitive explanation is that in state-independent encoding, each codeword must satisfy the dc^K-free constraint (dc^K-free word), whereas the state-dependent dc^K-free encoding releases the Kth-order spectral-null constraint for individual codewords. A larger number of codewords can be used in state-dependent encoding than in state-independent encoding. From expression (2.31), the supremum asymptotic redundancy of K-OZD codes is $O(K^2)$ implying that the relaxed spectral-null constraints may result in a significant increase in the number of available codewords. Furthermore, in Chapter 5 of this thesis, it will be shown that a state-dependent fixed-length encoder can be modeled as an equivalent state-independent variable-length encoder, and that given the word length and spectral-null constraint, state-dependent fixed-length HOSN codes may result in better spectrum performance than state-independent fixed-length HOSN codes.

Figure 3.2 illustrates the FSTD G and 4th-extension of G for the 3-state 2/4 state-dependent dc^2-free code constructed in [26]. In Figure 3.2 (b), the large circles represent the principal states. Figures 3.1 and 3.2 demonstrate that given a codeword length and spectral-null constraint, state-dependent encoding can result in an increased code rate compared to state-independent encoding.
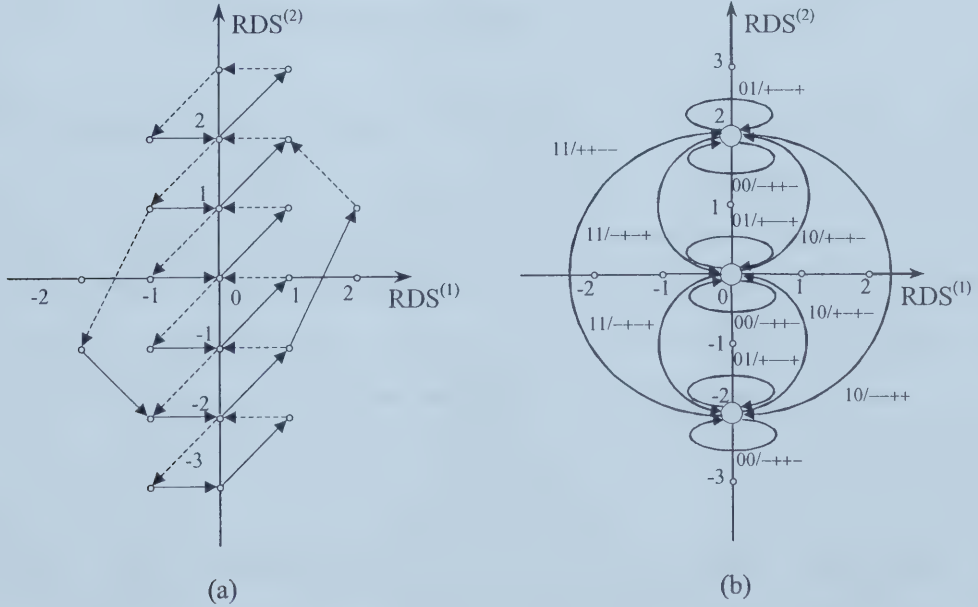
Figure 3.2   (a) FSTD of the 3-state 2/4 dc^2-free code G (solid lines and dashed lines denote $+1$ and -1 respectively); (b) 4th-extension of the FSTD for the 2/4 dc^2-free code G^4.

As described above, in practice a coded sequence with a high-order spectral null at dc is constructed through concatenation of codewords. To analyze such a block code, knowledge of the power spectrum of the code is necessary [1, ch.9-11, ch.13-14], [61], [62]. Alternatively, to construct a block coded high-order spectral-null sequence with large rejection at low frequencies, it is desirable to understand the relationship between the power spectrum of the code and other characteristics. Evaluation of the power spectra of fixed-length codes and state-independent variable-length codes is considered below.

3.1.2 Power Spectra of Fixed-Length Codes

A. Power spectra of state-dependent fixed-length codes

After the pioneering work [36] on the spectrum evaluation of coded signals, calculation of the power spectra of block coded sequences was discussed in [64], [65]. By considering the FSSM model of practical encoders, a general method for evaluating power spectra of block coded sequences, which is currently widely used, was developed in [66], [34]. The discussion in this subsection is based on this well-known method.

Let the encoded symbol sequence $\{x\} = \{x_1, x_2, \cdots, x_l, \cdots\}$ be generated through concatenation of length-N codewords $\{\underline{x}\} = \{\underline{x}_1, \underline{x}_2, \cdots, \underline{x}_n, \cdots\}$ where $\underline{x}_n = [x_{n,1}, x_{n,2}, \cdots, x_{n,N}]$ is as defined in Section 3.1.1 and the duration of each coded symbol is T_b. Let

$$x_{(n-1)N+j} = x_{n,j}. \tag{3.1}$$

The word sequence $\{\underline{x}\}$ is an N-dimensional discrete real-valued stochastic process. Assume that the symbols in source sequence $\{y\}$ are independent identically distributed (i.i.d.), i.e., stationary and independent from one symbol interval to the next. As a result, the source words are also independent identically distributed. Similar to symbol-by-symbol encoders described in Section 2.2.2, it can be shown that the coded word sequence $\{\underline{x}\}$, the output of the FSSM $\mathcal{E} = (\mathbb{Y}^M, \mathbb{X}^N, \mathcal{S}, g, h)$, is a wide-sense stationary process [34], [46] with:

$$\boldsymbol{m}_{\underline{x}} = E\{\underline{x}_n\} \tag{3.2}$$

$$\boldsymbol{R}_{\underline{x}}(kT) = E\{\underline{x}'_n \underline{x}_{n+k}\} \tag{3.3}$$

where $\boldsymbol{m}_{\underline{x}}$ is an N-dimensional row vector, $\boldsymbol{R}_{\underline{x}}(kT)$ is an $N \times N$ matrix ($\boldsymbol{R}_{\underline{x}}(kT) = \left\| R_{\underline{x}}^{p,q}(kT) \right\|$, ($p, q = 1, \cdots, N$), and $T = NT_b$. The prime represents transposition in this thesis. Therefore, the coded symbol sequence $\{x\}$ is a wide-sense cyclostationary process [66] with a period of T where:

$$E\{x_l\} = m_x((n-1)T + jT_b) = m_x(jT_b)$$ (3.4)

$$E\{x_i x_{l+m}\} = R_x((n-1)T + jT_b, mT_b) = R_x(jT_b, mT_b),$$ (3.5)

where $j = 1, \cdots, N$ and $l = (n-1)T + jT_b$ ($n \in \mathcal{I}$).

The power spectral density function of a stationary stochastic process is defined as the Fourier transform of the autocorrelation function [35]. The power spectrum of the coded word sequence $\{\underline{x}\}$ is:

$$H_{\underline{x}}(\omega) = \sum_{k=-\infty}^{\infty} R_{\underline{x}}(kT)e^{-ik\omega T}$$ (3.6)

where $i = \sqrt{-1}$. Since $\{\underline{x}\}$ is real, $R_{\underline{x}}(kT)$ is real and even and $H_{\underline{x}}(\omega)$ is real, even, and non-negative [35, ch.10]. Therefore, $R_{\underline{x}}(kT)$ as well as $H_{\underline{x}}(\omega)$ is a symmetric matrix.

A cyclostationary stochastic process, which is not a stationary stochastic process, cannot be directly characterized with a power spectrum. The spectral analysis of a cyclostationary process is carried out by evaluating the power spectrum of the corresponding stationary process which has the same mean and autocorrelation values as the average mean and average autocorrelation values of the cyclostationary process averaged over a period, or equivalently, as the mean and autocorrelation values of the phase-randomized process where the time reference or phase of the cyclostationary process is a random variable uniformly distributed over a period [38]. The average mean and autocorrelation are:

$$\overline{m}_x = \frac{1}{N} \sum_{j=1}^{N} m_x(jT_b)$$ (3.7)

$$\overline{R}_x(mT_b) = \frac{1}{N} \sum_{j=1}^{N} R_x(jT_b, mT_b).$$ (3.8)

As defined in [34], the average power spectrum of the coded symbol sequence $\{x\}$, $H_x(\omega)$, is the Fourier transform of the average autocorrelation:

$$H_x(\omega) = \sum_{m=-\infty}^{\infty} \overline{R}_x(mT_b)e^{-im\omega T_b}.$$ (3.9)

The relation between the average autocorrelation $\overline{R}_x(mT_b)$ and the autocorrelation of the word sequence $R_{\underline{x}}(kT)$ is derived below. From the definition (3.1), there is:

$$R_{\underline{x}}^{p,q}(kT) = E\{x_{n,p} x_{n+k,q}\}$$

$$= E\{x_{(n-1)N+p} x_{(n+k-1)N+q}\}$$

$$= E\{x_{(n-1)N+p} x_{(n-1)N+p+kN+q-p}\}$$

$$= R_x((n-1)NT_b + pT_b, (kN + q - p)T_b).$$ (3.10)

Let $h = q - p$ ($0 \le h \le N-1$). When $1 \le p \le N-h$, replacing $h = q - p$ into (3.10) yields:

$$R_x((n-1)T + pT_b, kT + hT_b) = R_{\underline{x}}^{p,p+h}(kT). \tag{3.11}$$

Similarly,

$$R_{\underline{x}}^{p,q}(kT + T) = E\{x_{n,p}x_{n+k+1,q}\} = R_x((n-1)NT_b + pT_b, (kN + N + q - p)T_b). \tag{3.12}$$

Let $h = N + q - p$ ($0 \le h \le N-1$). When $N + 1 - h \le p \le N$, replacing $h = N + q - p$ into (3.12) yields:

$$R_x((n-1)T + pT_b, kT + hT_b) = R_{\underline{x}}^{p,p+h-N}(kT + T). \tag{3.13}$$

From (3.5), (3.8), (3.11), and (3.13), the average autocorrelation is obtained:

$$\bar{R}_x(kT + hT_b) = \frac{1}{N}\sum_{p=1}^{N} R_x((n-1)T + pT_b, kT + hT_b)$$

$$= \frac{1}{N}\left[\sum_{p=1}^{N-h} R_{\underline{x}}^{p,p+h}(kT) + \sum_{p=N+1-h}^{N} R_{\underline{x}}^{p,p+h-N}(kT + T)\right]. \tag{3.14}$$

Relation (3.14) is illustrated in Figure 3.3 for the case of $N = 4$ [34].
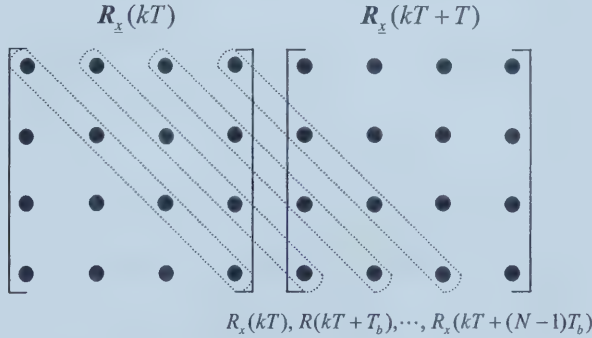
Figure 3.3   Visualization of the relationship between the average autocorrelation of symbol sequence and the autocorrelation of the word sequence ($N = 4$).

From (3.9) and (3.14), the relation between $H_x(\omega)$ and $H_{\underline{x}}(\omega)$ is obtained:

$$H_x(\omega) = \frac{1}{N}V^*(\omega)H_{\underline{x}}(\omega)V(\omega), \tag{3.15}$$

where $V(\omega) = [e^{-i\omega T_b}, \cdots, e^{-i\omega NT_b}]'$ and $V^*(\omega)$ is the conjugate transpose of $V(\omega)$. The spectral analysis of a coded symbol sequence is performed by evaluating the power spectrum of the coded word sequence. Since $H_{\underline{x}}(\omega)$ is a symmetric matrix, it can be shown that $H_x(\omega)$ is even.

In general, the power spectra $H_{\underline{x}}(\omega)$ and $H_x(\omega)$ each contains two components [34], [66]: a continuous component and a discrete component. The continuous component is:

$$H_{\underline{x}}^{(c)}(\omega) = \sum_{k=-\infty}^{\infty} \left[R_{\underline{x}}(kT) - R_{\underline{x}}(\infty) \right] e^{-ik\omega T}$$

$$= \sum_{k=-\infty}^{\infty} \left[R_{\underline{x}}(kT) - m'_{\underline{x}} m_{\underline{x}} \right] e^{-ik\omega T} \,. \tag{3.16}$$

From (3.15), there is:

$$H_x^{(c)}(\omega) = \frac{1}{N} V^*(\omega) H_{\underline{x}}^{(c)}(\omega) V(\omega) \,. \tag{3.17}$$

The discrete component is:

$$H_{\underline{x}}^{(d)}(\omega) = \sum_{k=-\infty}^{\infty} R_{\underline{x}}(\infty) e^{-ik\omega T} = \sum_{k=-\infty}^{\infty} m'_{\underline{x}} m_{\underline{x}} e^{-ik\omega NT_b} \,. \tag{3.18}$$

It has been given [67, pp. 271-272] that:

$$\sum_{k=-\infty}^{\infty} e^{-i\omega k N} = \sum_{n=-\infty}^{\infty} \delta(n-kN) e^{-i\omega n} = (2\pi/N) \sum_{k=-\infty}^{\infty} \delta(\omega - 2\pi k/N) \,,$$

where $\delta(\cdot)$ represents the impulse function. Therefore

$$H_{\underline{x}}^{(d)}(\omega) = m'_{\underline{x}} m_{\underline{x}} \sum_{k=-\infty}^{\infty} (2\pi/N) \delta(\omega - 2\pi k/(NT_b)) \,. \tag{3.19}$$

Also from (3.15), it can be obtained that

$$H_x^{(d)}(\omega) = \frac{1}{N} V^*(\omega) H_{\underline{x}}^{(d)}(\omega) V(\omega)$$

$$= \frac{1}{N^2} \sum_{k=-\infty}^{\infty} V^*(\omega) m'_{\underline{x}} m_{\underline{x}} V(\omega) 2\pi \delta(\omega - 2\pi k/(NT_b)) \,. \tag{3.20}$$

Let the mean vector of the coded word sequence be $m_{\underline{x}} = [m_1, \cdots, m_N]$ and

$$M_k = m_{\underline{x}} V(\omega) = \sum_{j=1}^{N} m_j e^{-i2\pi jk/(NT_b)} \,.$$

Therefore,

$$H_x^{(d)}(\omega) = = \frac{1}{N^2} \sum_{k=-\infty}^{\infty} |M_k|^2 \, 2\pi \delta(\omega - 2\pi k/(NT_b)) \,. \tag{3.21}$$

The equation (3.21) demonstrates that when the values of $|M_k|^2$ are non-zero, there are spectral lines at the multiples of the frequency $1/NT_b$.

Expressions (3.16)-(3.21) show that the power spectrum of the coded symbol sequence is determined by both the mean and autocorrelation of the corresponding coded word sequence. Let a block encoder be modeled as an FSSM $\mathcal{E} = (\mathbb{Y}^M, \mathbb{X}^N, \mathcal{S}, g, h)$ which is defined in Section 3.1.1. Assume that the binary input symbols of the encoder are independent identically distributed (i.i.d.). The state sequence $\{s_n\}$ is stationary [43]. The probability of the uth input

49

word $\beta_u \in \mathbb{Y}^M$ $(u = 1, \cdots, 2^M)$, q_u, equals the product of the probabilities of M symbols in this word, and is denoted $q_u = \Pr(\underline{y}_n = \beta_u)$. The state transition matrix $E_u = \left\| E_u^{a,b} \right\|$ is an $S \times S$ matrix with

$$E_u^{a,b} = \begin{cases} 1, & \text{if } g(\sigma_a, \beta_u) = \sigma_b \\ 0, & \text{otherwise} \end{cases} \tag{3.22}$$

where $a, b = 1, \cdots, S$ (S is the total state number). Therefore, the transition probability matrix of the Markov chain in FSSM, Q, which is also an $S \times S$ matrix, is obtained:

$$Q = \sum_{u=1}^{2^M} q_u E_u . \tag{3.23}$$

Assume that the Markov chain is irreducible and aperiodic [34]. The limiting transition probability matrix exists [68, p.152] such that:

$$Q_\infty = \lim_{k \to \infty} Q^k = \mathbf{w}\,\pi \tag{3.24}$$

where $\mathbf{w}$ is an S-dimensional all-one column vector and $\pi = [\pi_1, \cdots, \pi_S]$ is the steady-state probability vector.

Let the output word matrix corresponding to the uth source word be Γ_u of dimension $S \times N$. For the Mealy model, the sth row of Γ_u is the codeword $\alpha_{su} = h(\sigma_s, \beta_u) \in \mathbb{X}^N$, $s = 1, \cdots, S$ and the codeword mean and autocorrelation are given by [34]:

$$m_{\underline{x}} = \sum_{u=1}^{2^M} q_u \pi \Gamma_u \tag{3.25}$$

$$R_{\underline{x}}(kT) = \begin{cases} \displaystyle\sum_{u=1}^{2^M} q_u \Gamma_u' D \Gamma_u, & k = 0 \\ \displaystyle\sum_{u,v=1}^{2^M} q_u q_v \Gamma_u' D E_u Q^{k-1} \Gamma_v, & k \geq 1. \end{cases} \tag{3.26}$$

where D is the diagonal matrix of state probabilities, i.e., $D = \mathrm{diag}(\pi_1, \cdots, \pi_S)$. Using the notation defined in [34], the continuous component of power spectrum of the word sequence is given by the theorem below.

Theorem 3.1 [34]: $H_{\underline{x}}^{(c)}(\omega) = Y(z) + Y(z^{-1})$, $z = e^{i\omega T}$, where

$$Y(z) = (1/2)\left(C_0 - m_{\underline{x}}' m_{\underline{x}}\right) + C_1 B(z) C_2 \tag{3.27}$$

with

$$C_0 = \sum_{u=1}^{2^M} q_u \Gamma_u' D \Gamma_u$$

$$C_1 = \sum_{q=1}^{2^M} q_u \Gamma_u' DE_u (U - Q_\infty)$$

$$C_2 = \sum_{v=1}^{2^M} q_v \Gamma_v$$

$$B(z) = \left[zU - (Q - Q_\infty) \right]^{-1}$$

in which U is the identity matrix.

Power spectra of the 4-state 2/4 dc²-free code and the 3B/3T dc²-free code previously shown in Figures 2.1 and 2.2 were evaluated through the method introduced above. In the remainder of this thesis, spectral analysis for state-dependent fixed-length codes is based on this method. Spectral analysis for state-independent fixed-length codes, which is described below, is simpler than the case of state-dependent encoding.

B. Power spectra of state-independent fixed-length codes

Given the assumption of independent input source words, in a state-independent encoded sequence generated through concatenation of words with the same word length, the correlation between any two different codewords is zero, i.e., $R_x(kT)$ is an all-zero matrix when $k \geq 1$. Let $\underline{x}_a = [x_{a,1}, x_{a,2}, \cdots, x_{a,N}]$ be an element in the codeword set $\mathcal{A}$ with cardinality A_c. The discrete Fourier transform $X_a(\omega)$ of the codeword $\underline{x}_a$ is:

$$X_a(\omega) = \sum_{j=1}^{N} x_{a,j} e^{-ij\omega T_b} . \tag{3.28}$$

Assume that A $(A \leq A_c)$ words are used as codewords, that they are equiprobable, and that the mean of the words m_x is the all-zero vector. From (3.20), the discrete component of power spectrum $H_x^{(d)}(\omega)$ is zero, i.e., there are no spectral lines in the spectrum. From (3.16) and (3.17), it can be verified that the continuous component of power spectrum $H_x^{(c)}(\omega)$ is [23]:

$$H_x^{(c)}(\omega) = \frac{1}{N} V^*(\omega) R_x(0) V(\omega)$$

$$= \frac{1}{N} V^*(\omega) E\{\underline{x}_a' \underline{x}_a\} V(\omega)$$

$$= \frac{1}{N} [e^{i\omega T_b}, \cdots, e^{i\omega N T_b}] E\{\underline{x}_a' \underline{x}_a\} [e^{-i\omega T_b}, \cdots, e^{-i\omega N T_b}]'$$

$$= \frac{1}{AN} \sum_{a=1}^{A} |X_a(\omega)|^2 . \tag{3.29}$$

If the codewords are known, by means of (3.29), it is convenient to obtain the power spectra of high-order zero-disparity codes defined in Section 2.3.

Example 3.1:

a) For 1/2 1-OZD code, i.e., bi-phase code (codewords: "$+-$" and "$-+$"),

$$H_{x1}(\omega) = 2\sin^2(\omega T_b/2);\qquad\qquad(3.30)$$

b) For 1/4 2-OZD code (codewords: "$+--+$" and "$-++-$"),

$$H_{x2}(\omega) = 4\sin^2(\omega T_b/2)\sin^2(\omega T_b);\qquad\qquad(3.31)$$

c) For 1/8 3-OZD code (codewords: "$+--+-++-$" and "$-++-+--+$"),

$$H_{x3}(\omega) = 8\sin^2(\omega T_b/2)\sin^2(\omega T_b)\sin^2(2\omega T_b).\qquad\qquad(3.32)$$

Note that $H_{x1}(\omega)$ and $H_{x2}(\omega)$ have also been derived in [69] and [23] respectively. In Figure 3.4, power spectra of the 1/2 1-OZD, 1/4 2-OZD, and 1/8 3-OZD codes are plotted.
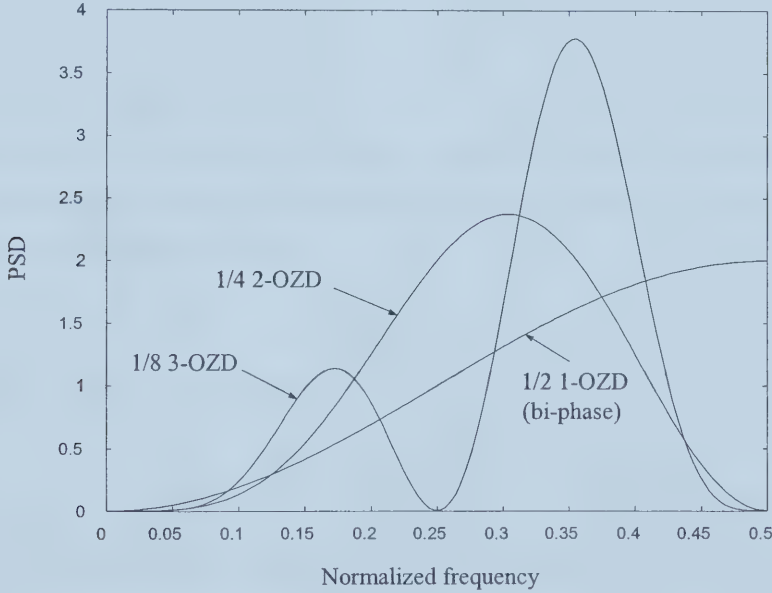
Figure 3.4 Power spectra of the 1/2 1-OZD, 1/4 2-OZD, 1/8 3-OZD codes.

C. *Power spectra of K-OZD codes*

Although expression (3.29) can be used to evaluate the power spectrum of any K-OZD codes, due to the exponential growth of the number of codewords with the increase of word length [29], when the codeword length is large, the analytic method becomes prohibitive. However, consideration of full-set coding and the enumeration method performed in Section 2.3.3 can reduce the computational complexity and allow the evaluation of further results.

Average power spectra of K-OZD codes can be evaluated by considering the full set of words $\mathcal{A}(N, K)$, i.e., all available dc^K-free words are used as codewords with equal probability. In this

case, the number of codewords A equals the cardinality of $\mathcal{A}(N, K)$, A_c. This kind of coding is called *full-set coding*. Most block codes generated through full-set coding are nonconstructive because generally the cardinality of a codeword set does not equal a power of two. However, full-set coding can make the spectral analysis of K-OZD codes more convenient.

From (3.9) and (3.14), the power spectrum of a state-independent encoded sequence can also be expressed as:

$$H_x(\omega) = \sum_{k=N-1}^{-(N-1)} \overline{R}_x(kT_b)e^{-ik\omega T_b}$$

(3.33)

with

$$\overline{R}_x(hT_b) = (1/N)\sum_{j=1}^{N-h} E(x_{a,j}x_{a,j+h}), \ 1 \le h \le N-1$$

$$\overline{R}_x(0) = 1$$

$$\overline{R}_x(hT_b) = 0, h \ge N .$$

Power spectra of 1-OZD codes considering the full set of dc^1-free words with given word length were developed in [69]. As shown in [69], the correlation of symbols at different positions within the same codeword is:

$$r_0 = E\left\{x_{a,j_1}x_{a,j_2}\right\} = -\frac{1}{N-1}, \ j_1 \ne j_2, 1 \le j_1, j_2 \le N .$$

Combining (3.33) and r_0 above yields:

$$H_x(\omega) = \frac{N}{N-1}\left[1 - \frac{1}{N^2}\left(\frac{\sin(N\omega T_b/2)}{\sin(\omega T_b/2)}\right)^2\right].$$

(3.34)

Figure 3.5 illustrates the power spectra of 1-OZD codes generated by full-set coding at low frequencies.

For K-OZD codes ($K \ge 2$), evaluation of the correlation r_0 can be accomplished by employing the enumeration method. The algorithm described below is suitable for evaluation of power spectra of arbitrarily high-order zero disparity codes, which is an extension of the algorithm for the case of 2-OZD codes introduced in [22].

Because each dc^K-free word $\underline{x}_a \in \{-1, 1\}^N$, there is:

$$r_0 = E\left\{x_{a,j_1}x_{a,j_2}\right\} = (1/A)\left[A(x_{a,j_1} = x_{a,j_2}) - A(x_{a,j_1} \ne x_{a,j_2})\right], \ j_1 \ne j_2 ,$$

where $A(\cdot)$ represents the number of codewords satisfying the condition in the parenthesis.
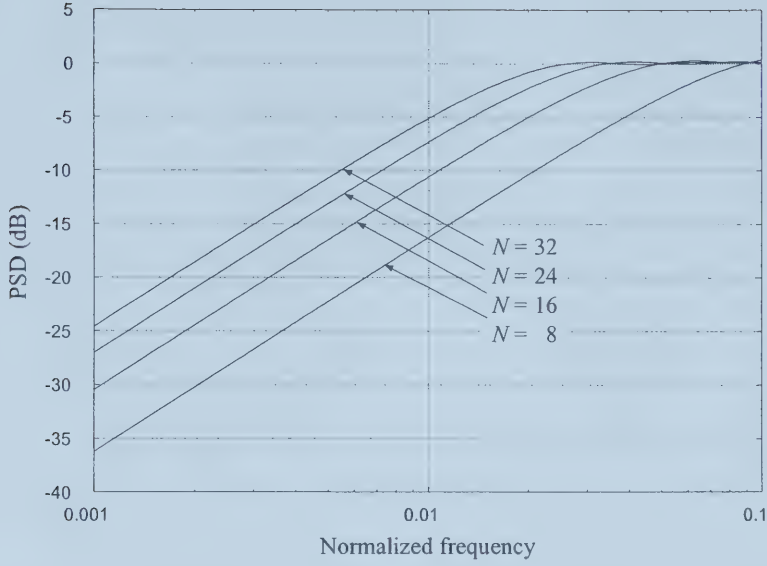
Figure 3.5 Power spectra of 1-OZD codes generated by full-set coding.

Furthermore, due to the symmetry [22]

$$r_0 = 4(1/A)A(x_{a,j_1} = x_{a,j_2} = 1) - 1. \tag{3.35}$$

Let $\alpha = \{\alpha_1, \cdots, \alpha_{N/2}\}$ be the indices of the word $\underline{x}_a$ in which $x_{a,\alpha_p} = +1$ is as defined in Section 2.3.1. Therefore

$$\sum_{p=1, \alpha_p \neq j_1, j_2}^{N/2} \alpha_p^r = (1/2) \sum_{j=1}^{N} j^r - j_1^r - j_2^r = J_r - j_1^r - j_2^r, r = 0, 1, \cdots, K-1.$$

Similar to (2.34), another K-variable generating function can be constructed to evaluate $A(x_{a,j_1} = x_{a,j_2} = 1)$:

$$\frac{F_N(v_0, v_1, \cdots, v_{K-1})}{(1 + v_0^{j_1^0} v_1^{j_1^1} \cdots v_{K-1}^{j_1^{(K-1)}})(1 + v_0^{j_2^0} v_1^{j_2^1} \cdots v_{K-1}^{j_2^{(K-1)}})} = \prod_{j=1, j \neq j_1, j_2}^{N} (1 + v_0^{j^0} v_1^{j^1} \cdots v_{K-1}^{j^{(K-1)}})$$

$$= \sum_{\lambda_0, \lambda_1, \cdots, \lambda_{K-1}} \Lambda_N(\lambda_0, \lambda_1, \cdots, \lambda_{K-1}) \cdot v_0^{\lambda_0} v_1^{\lambda_1} \cdots v_{K-1}^{\lambda_{K-1}} . \tag{3.36}$$

When $\lambda_r = J_r - j_1^r - j_2^r, r = 0, 1, \cdots, K-1$, a valid word is enumerated. A similar recursion to (2.35) can also be derived with initial conditions

$$\Lambda_0(0, 0, \cdots, 0) = 1, \text{ and } \Lambda_0(\lambda_0, \lambda_1, \cdots, \lambda_{K-1}) = 0 \text{ for } \lambda_i \neq 0, i = 0, 1, \cdots, K-1.$$

Based on the algorithm introduced above, power spectra of 2-OZD, 3-OZD, and 4-OZD codes can be evaluated, and are illustrated in Figures 3.6 – 3.8.
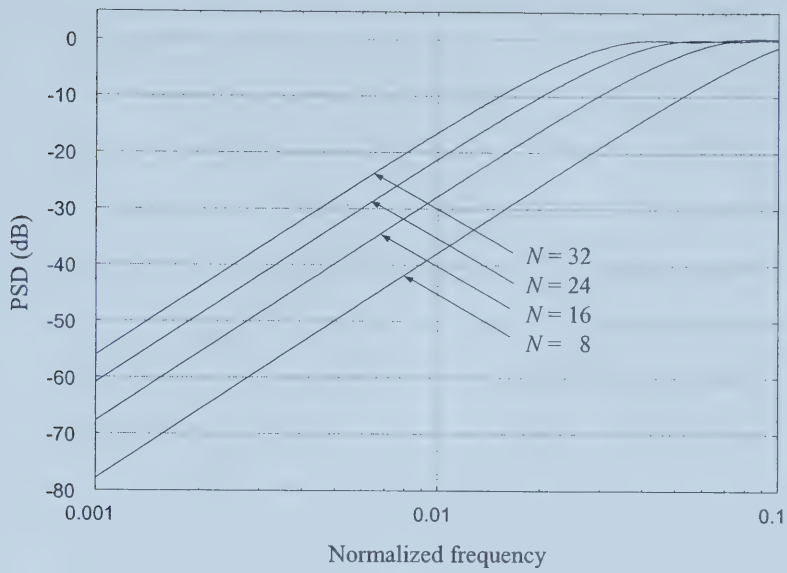
Figure 3.6 Power spectra of 2-OZD codes generated through full-set coding.
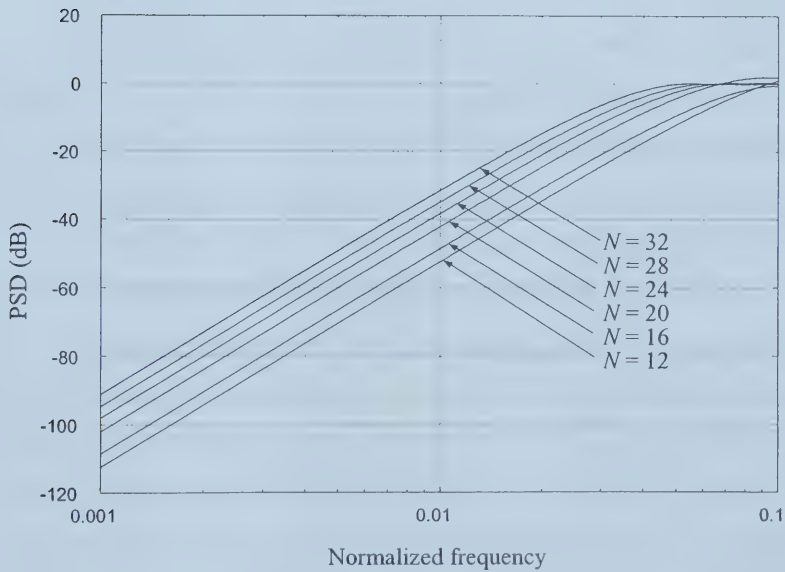
Figure 3.7 Power spectra of 3-OZD codes generated through full-set coding.
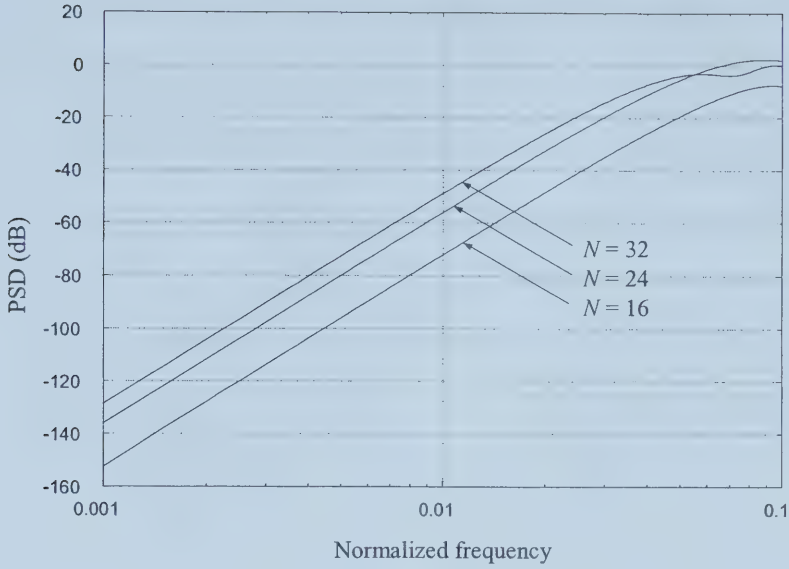
Figure 3.8 Power spectra of 4-OZD codes generated through full-set coding.

3.1.3   Power Spectra of Variable-Length Codes

Evaluation of the power spectra of variable-length codes has been extensively developed in [70], [71]. A variable-length symbol sequence is generated through concatenation of words with variable lengths, and the separation between two consecutive symbols is defined as T_b. The content discussed in this subsection will be applied to evaluation of low-frequency characteristics of state-dependent fixed-length dc^K-free sequences $(K \geq 1)$ in Chapter 5 of this thesis where a state-dependent fixed-length dc^K-free encoder is equivalently modeled as a state-independent variable-length dc^K-free encoder. For this application, only the continuous part of the PSD of state-independent variable-length codes is considered in this subsection.

Compared to the case of fixed-length codes, evaluation of power spectra of variable-length codes is more complicated. In addition to the coded word/symbol sequence, the word length sequence itself is also a stochastic process. Assume that $\mathcal{B}$ is a finite set of variable-length words with lengths from a finite set of word lengths $\mathcal{L}$ in which all elements are positive integers. Let $\underline{B}_m$ be a vector of dimension L_m of the mth word, i.e., $\underline{B}_m = [x_{m,1}, x_{m,2}, \cdots, x_{m,L_m}]$, and let an infinite sequence of variable-length words be $\{\underline{B}\} = \{\underline{B}_m, \underline{B}_m \in \mathcal{B}\}$; $\{L_m, L_m \in \mathcal{L}\}$ is the corresponding sequence of variable word lengths.

Similar to fixed-length encoders, assume that in variable-length encoders the symbols in source sequence are i.i.d.. Therefore, the source words are also independent identically

distributed. It can be shown that the coded word sequence $\{\underline{B}_m\}$, the output of the FSSM, is wide-sense stationary [70]. Denote the probabilities of length-λ words as $p_\lambda = \Pr\{L_m = \lambda\}$, $\lambda \in \mathcal{L}$. Then the mean word length is: $L = \sum_\lambda \lambda p_\lambda$. Let $\mathcal{B}_\lambda = \{\underline{\beta}_\lambda(1), \underline{\beta}_\lambda(2), \cdots, \underline{\beta}_\lambda(\eta_\lambda)\}$ be a set of η_λ words with length λ where $\underline{\beta}_\lambda(s) = [x_{\lambda 1}(s), x_{\lambda 2}(s), \cdots, x_{\lambda\lambda}(s)]$, $\underline{\beta}_\lambda(s) \in \mathcal{I}^\lambda$ (where $\mathcal{I}$ is the set of integers) and $s = 1, \cdots, \eta_\lambda$, and let $\beta_\lambda = \|x_{\lambda j}(s)\|$ be an $\eta_\lambda \times \lambda$ matrix where $j = 1, \cdots, \lambda$, $s = 1, \cdots, \eta_\lambda$, and $\lambda \in \mathcal{L}$. Also let $p_\lambda(s) = \Pr\{\underline{B}_m = \underline{\beta}_\lambda(s)\}$ and $P_\lambda = [p_\lambda(1), \cdots, p_\lambda(\eta_\lambda)]$. Due to the variability of the word lengths in $\{\underline{B}_m\}$, the mean value of $\{\underline{B}_m\}$ cannot directly be given as it was in (3.2) for the fixed-length case. It is conditional on the word length λ, i.e.,

$$m_\lambda = E\{\underline{B}_m; L_m = \lambda\} = \sum_s p_\lambda(s)\underline{\beta}_\lambda(s) = P_\lambda \beta_\lambda \qquad (3.37)$$

which is a vector of dimension λ, and represents the mean value of the words of length λ in $\{\underline{B}_m\}$. For fixed-length coding, the autocorrelation between codewords is expressed in (3.3) where the separation is a multiple of word length T. However, for variable-length coding, even if the number of codewords between two codewords is the same, the time separation could be different. An example is shown in Figure 3.9. There is one codeword between $\underline{B}_{-2}$ and $\underline{B}_0$, and between $\underline{B}_{-1}$ and $\underline{B}_1$. However, the digits' distances are 8 and 10 respectively, where it is assumed that each digit occupies the same symbol duration T_b. Instead of taking into account the distance in terms of number of words, analysis of a variable-length word sequence considers the distance between two codewords in terms of the number of digits separation.
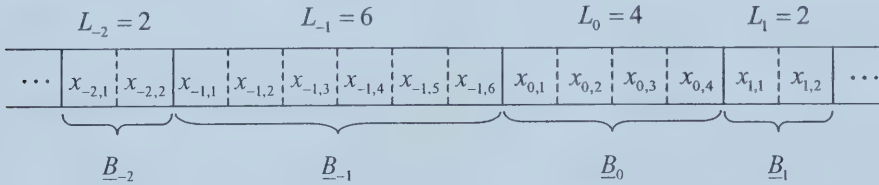
Figure 3.9 Illustration of an example variable-length symbol sequence.

Denote the digit distance between two codewords $\underline{B}_{m1}$ and $\underline{B}_{m2}$ as $\tau_{m1,m2}$. Assume that the word distance of these codewords is $m1 - m2 = l$. There is:

$$
\tau_{m1,m2} = \begin{cases} \displaystyle\sum_{m=m1}^{m2-1} L_m, & m2 > m1 \\ 0, & m2 = m1 \\ -\tau_{m1,m2}, & m2 < m1 . \end{cases} \tag{3.38}
$$

Let the cumulative length probabilities be:

$$
y(k) = \begin{cases} \Pr\left\{ \displaystyle\bigcup_{l=0}^{\infty} (\tau_{m1,m1+l} = k) \right\}, & \text{if } k \geq 0 \\ 0, & k < 0 \end{cases} \tag{3.39}
$$

Therefore, joint probabilities between two words in order with lengths λ and μ at k digit distance are arranged in an $\eta_\lambda \times \eta_\mu$ matrix as shown in (40) of [70]:

$$
\boldsymbol{P}_{\lambda\mu}(k) = \begin{cases} \delta_{\lambda\mu} \mathrm{diag}\left[p_\lambda(1), \cdots, p_\lambda(\eta_\lambda) \right], & k = 0 \\ y(k-\lambda)\boldsymbol{P}_\lambda'\boldsymbol{P}_\mu, & k > 0 \\ y(-k-\mu)\boldsymbol{P}_\lambda'\boldsymbol{P}_\mu & k < 0 \end{cases} \tag{3.40}
$$

where $\delta_{\lambda\mu}$ is the Kronecker delta function. Let the greatest common divisor (g.c.d.) of word lengths in the set $\mathcal{L}$ be Λ. It has been shown (Remark 1 in [70]) that the quantities of $\boldsymbol{P}_{\lambda\mu}(k)$ and $y(k)$ require that k must be an integer multiple of Λ since the digit distance should count for all digits between two codewords. Denote $k = h\Lambda$, $h = 0, 1, \cdots$. The condition when $k \neq 0$ in (3.40) implies that $h \neq 0$, therefore the digit distance should not include the digits of the end word when considering the distance between two words (see (3.38)). Finally, the expressions for $\boldsymbol{P}_{\lambda\mu}(k)$ for the case of $k \neq 0$ are obtained as shown in (3.40).

As described in the subsection above, the PSD of a sequence can be considered in two parts: the continuous component and the discrete component. The continuous component of a variable-length word sequence is defined as [70]:

$$
\boldsymbol{H}_{\lambda\mu}(z) = \sum_{h=-\infty}^{\infty} \left[\boldsymbol{R}_{\lambda\mu}(h\Lambda T_b) - \boldsymbol{R}_{\lambda\mu}(\infty\Lambda T_b) \right] z^{-h\Lambda} . \tag{3.41}
$$

where $z = e^{i\omega T_b}$ and

$$
\boldsymbol{R}_{\lambda\mu}(h\Lambda T_b) = \boldsymbol{\beta}_\lambda' \boldsymbol{P}_{\lambda\mu}(h\Lambda) \boldsymbol{\beta}_\mu
$$

$$
\boldsymbol{R}_{\lambda\mu}(\infty\Lambda T_b) = \boldsymbol{\beta}_\lambda' \boldsymbol{P}_{\lambda\mu}(\infty\Lambda) \boldsymbol{\beta}_\mu .
$$

Assume that the words in the word sequence $\{\underline{B}_m\}$ are mutually independent. Therefore, the word lengths in the sequence $\{L_m\}$ are also mutually independent and the sequence $y(h\Lambda)$, $h = 0, 1, \cdots$, converges to $y(\infty\Lambda) = \Lambda/L$. From (3.40), it is convenient to obtain that:

$$P_{\lambda\mu}(\infty\Lambda) = y(\infty\Lambda)P_\lambda'P_\mu = (\Lambda/L)P_\lambda'P_\mu . \tag{3.42}$$

According to (3.37) and (3.40)-(3.42), there are:

$$R_{\lambda\mu}(kT_b) = \begin{cases} \delta_{\lambda\mu}\beta_\lambda'\mathrm{diag}[p_\lambda(1), \cdots, p_\lambda(\eta_\lambda)]\beta_\mu, & k = 0 \\ y(k-\lambda)m_\lambda'm_\mu, & k > 0 \\ y(-k-\mu)m_\lambda'm_\mu, & k < 0 \end{cases} \tag{3.43}$$

$$R_{\lambda\mu}(\infty\Lambda T_b) = y(\infty\Lambda)m_\lambda'm_\mu = (\Lambda/L)m_\lambda'm_\mu \tag{3.44}$$

where $m_\lambda = P_\lambda\beta_\lambda$ defined in (3.37). From (3.41)-(3.44), the following theorem results.

Theorem 3.2: If the matrix function $R_{\lambda\mu}(h\Lambda T_b)$ converges to $R_{\lambda\mu}(\infty\Lambda T_b)$, and the matrix function $R_{\lambda\mu}(h\Lambda T_b) - R_{\lambda\mu}(\infty\Lambda T_b)$ is absolute summable, then the continuous component of PSD of a variable-length word sequence with mutually independent words is given as:

$$H_{\lambda\mu}(z) = R_{\lambda\mu}(0) + m_\lambda'm_\mu\left[z^{-\lambda}X(z^\Lambda) + z^\mu X(z^{-\Lambda}) + (\Lambda/L)\left(1 - \sum_{h=0}^{\lambda'-1}z^{-h\Lambda} - \sum_{h=0}^{\mu'-1}z^{h\Lambda}\right)\right] \tag{3.45}$$

where

$$X(z) = \frac{(\Lambda/L)\sum_{\lambda'\geq 2} p_\lambda\left[(\lambda'-1)z^0 + (\lambda'-2)z^{-1} + \cdots + z^{-(\lambda'-2)}\right]}{\sum_\lambda p_\lambda(1 + z^{-1} + \cdots + z^{-(\lambda'-1)})}$$

and $\lambda' = \lambda/\Lambda$, $\mu' = \mu/\Lambda$.

Note that the expression for $X(z)$ in [70] contains typographical errors. Proof of Theorem 3.2 is deferred to Appendix 1.

Deframing is a procedure that converts a word sequence $\{\underline{B}_m\}$ into a symbol sequence. For variable-length encoding, even if $\{\underline{B}_m\}$ is stationary, after deframing, the symbol sequence in general is nonstationary when the time origin of the symbol sequence is deterministic [70]. To recover the stationarity, a symbol sequence with a random origin was considered in [70]. The continuous component of the power spectrum of such a symbol sequence associated with the continuous component of the power spectrum of corresponding word sequence was given in Theorem 6 of [70]:

$$H_x(z) = \frac{1}{L} V_\lambda(z) H_{\lambda\mu}(z) V_\mu^*(z) \tag{3.46}$$

where $H_{\lambda\mu}(z)$ is given by (3.45), L is the mean word length, $V_\lambda(z)$ is a row vector of dimension λ $(V_\lambda(z) = [z, \cdots, z^\lambda])$, and $V_\lambda^*(z)$ is the conjugate transpose of $V_\lambda(z)$.

3.2 Performance Metric for First-Order Spectral-Null Codes — Sum Variance

As discussed in Section 1.2, one of the most important characteristics of a code with a spectral null at zero frequency is its ability to suppress low-frequency components. Analysis of the exact power spectrum at low frequencies is not straightforward [18], however, to permit performance assessment and design of dc^K-free codes $(K \geq 1)$, low-frequency properties of dc^K-free codes need to be determined. For convenient analysis and design of a good code, it is also desirable that these properties in the frequency domain be associated with some time-domain parameters of the coded sequence.

For first-order spectral-null codes, heuristic methods for performance measurement which are related to RDS values of coded sequences, such as the variation of RDS and the variance of RDS, were employed with little justification [44], [72], [73] before the introduction of the sum variance metric in [62].

For a symbol sequence $\{x_l\}$ $(x_l \in \mathcal{I}$, $l = 1, 2, \cdots)$ with a first-order spectral null at dc, let $\text{RDS}_l^{(1)}$ be as defined in (2.7). Assume that $x_0 = 0$. Therefore $\text{RDS}_l^{(1)} = x_1 + \cdots + x_l$. The $\text{RDS}^{(1)}$ value of $\{x_l\}$ plays an important role in the analysis of dc^1-free codes. Capacity of a dc^1-free sequence as given in (2.9) is uniquely determined by the variation of $\text{RDS}^{(1)}$. Also, in [62] a well-known approximate relationship in dc^1-free codes was given:

$$2\sigma_{\text{RDS}^{(1)}}^2 \omega_0 \approx 1 \tag{3.47}$$

where $\sigma_{\text{RDS}^{(1)}}^2$ represents the value of $E\left[\left(\text{RDS}_l^{(1)}\right)^2\right]$, which is equivalent to the variance of $\text{RDS}_l^{(1)}$ when the mean value of $\text{RDS}_l^{(1)}$ is zero, and ω_0 is denoted as the cutoff frequency (Nyquist bandwidth) [69] at which the PSD value equals 0.5 or –3dB. It has been shown [1, p.219], [62] that approximation (3.47) is reliable and is accurate to within a few percent.

It is assumed in [62] that the power spectra of the considered codes remain small up to the cutoff frequency and then become approximately constant over the rest of the frequency band.

With this assumption, the cutoff frequency can be used as a measurement of the width of spectral null. Due to ease of evaluation for sum variance and the reliability of approximation (3.47) [1, p. 215], [62], sum variance has been widely used as a performance metric to evaluate the width of spectral null, and furthermore, the suppression of low-frequency components of the coded sequence [1, ch.9-10], [10], [18], [19], [21], [31], [74]-[76]. From (3.47), it is straightforward to observe that the smaller the value of sum variance, the wider the spectral null as well as the more spectrum suppression at low frequencies.

Evaluation of sum variance rather than evaluation of the entire power spectrum makes it possible to assess coded sequences analytically. Some results are described below.

- *Maxentropic dc^1-free codes*

Shannon [32] proved that given constraints, the code rate equals the capacity when the transition probabilities in the FSSM maximize the entropy of the coded sequences. By recalling (2.9), the capacity of a dc^1-free code is uniquely determined by the variation of RDS values of the coded sequence V [33]. The stationary probability distribution and sum variance of maxentropic dc^1-free codes are given by, respectively [62]:

$$\pi_\gamma = \frac{2}{V+1} \sin^2 \frac{\gamma\pi}{V+1}, \ \gamma = 1, \cdots, V \tag{3.48}$$

and

$$\sigma_{\text{RDS}^{(1)}}^2 (V) = \sum_{\gamma=1}^{V} \left(\frac{V+1}{2} - \gamma \right)^2 \pi_\gamma = \frac{2}{V+1} \sum_{\gamma=1}^{V} \left(\frac{V+1}{2} - \gamma \right)^2 \sin^2 \frac{\gamma\pi}{V+1}. \tag{3.49}$$

Expression (3.49) demonstrates that sum variance is also dependent only on the values of V. In Figure 3.10, the relationship between sum variance and the redundancy $1 - C(V)$ of maxentropic dc^1-free sequences is illustrated. Note that the values on the curve are available only when V is an integer. Figure 3.10 shows that a wide spectral null is obtained at the cost of code rate, and provides some insight into the trade-off between code rate and spectrum performance of dc^1-free codes. For large values of digital sum variation V, it has been shown that [1, p. 200], [61]:

$$C(V) \approx 1 - \frac{\pi^2}{2\ln 2} \frac{1}{(V+1)^2} \tag{3.50}$$

and

$$\sigma_{\text{RDS}^{(1)}}^2 (V) \approx \left(\frac{1}{12} - \frac{1}{2\pi^2} \right)(V+1)^2 + O\left(\frac{1}{(V+1)^2} \right). \tag{3.51}$$

The asymptotic product of sum variance and redundancy is approximately constant:

$$(1-C(V))\sigma^2_{\text{RDS}^{(1)}}(V) \approx \frac{\pi^2/6-1}{4\ln 2} = 0.2326, \quad V \gg 1. \tag{3.52}$$

It has been observed that when $N > 9$, the accuracy of approximation (3.52) is within 1%.
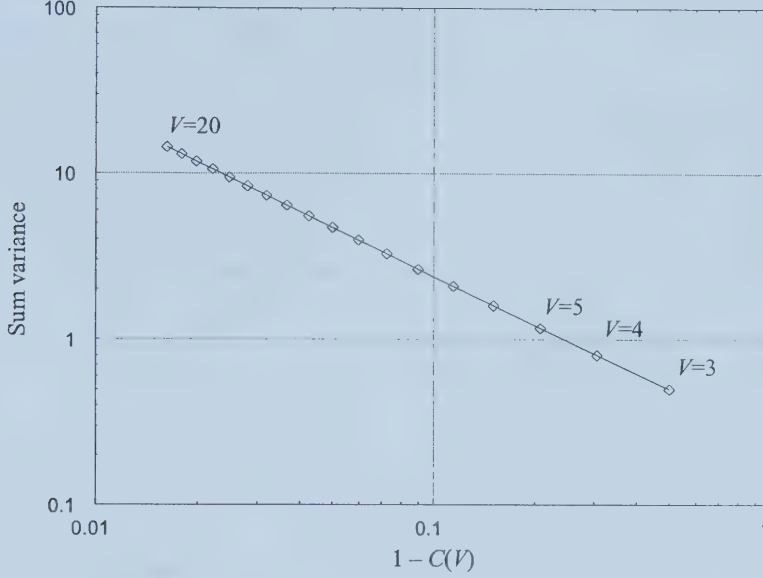
Figure 3.10 Illustration of sum variance versus redundancy
of maxentropic dc^1-free sequence.

Theorem 3.3 [62, Theorem 4]: The entropy of a binary dc^1-free sequence with sum variance $\sigma^2_{\text{RDS}^{(1)}}$ is upper bounded by the entropy of the maxentropic dc^1-free sequence with the same given sum variance value.

To compare alternative dc^1-free codes and maxentropic dc^1-free sequences, both coding rate and spectrum performance should be considered. The encoder efficiency has been defined as the ratio of redundancy-sum variance products of the maxentropic dc^1-free sequence and practical dc^1-free codes [61], i.e.,

$$E = \frac{[1-C(V)]\sigma^2_{\text{RDS}^{(1)}}(V)}{[1-R]s^2}. \tag{3.53}$$

From Theorem 3.3, it is straightforward to obtain that the encoder efficiency defined above cannot be larger than one.

- *First-order zero-disparity codes*

Kth-order zero-disparity codes have been defined in Section 2.3. Let the codeword length be N. For first-order zero-disparity codes, all binary codewords have an equal number of ones and

zeros. Some cutoff frequencies of first-order spectral-null codes for given relatively small values of N, which were obtained through measurement of PSD of 1-OZD codes, were tabulated in [69]. For 1-OZD codes generated through full-set coding where all available dc^1-free words are used, to evaluate the cutoff frequencies of these codes conveniently, a closed-form expression for sum variance of has been given [62]:

$$\sigma^2_{RDS^{(1)}} = (N+1)/6. \tag{3.54}$$

Note that the same expression as (3.54) has also been obtained in [61] with a different approach.

- *Bi-mode dc^1-free codes generated through full-set coding*

In a bi-mode dc^1-free code, one source word is mapped to two codewords that have different disparity. If all codewords with length N are used in bi-mode codes, the sum variance is given as [61]:

$$\sigma^2_{RDS^{(1)}} = \frac{1}{6}(5N-1) - \frac{(N+1)2^N}{12A} \tag{3.55}$$

where

$$A = 2^{N-1} + \frac{1}{2}\binom{N}{N/2}.$$

In advanced disk recording systems, the practical methods for accurate tracking are implemented by means of supplied servo position information that is recorded as low-frequency tones [10]. If track loss occurs during reading in the recording systems, error correction is useless [10], [20]. To realize "noise/interference free" tracking, the frequency content of the recorded data signals needs to be suppressed sufficiently at low frequencies compared to the channel bit rate, especially at the servo signal frequency band. For the DVD system [17], the reference channel bit rate is 26.16 Mb/s and the servo signal bandwidth is 20 kHz [31]. Therefore, the current most attractive consideration for a code with a spectral null at dc focuses on suppression of spectral content over the very low frequency band, e.g., at a normalized frequency of 10^{-3} and lower frequencies [10], [18]-[21], [31], [75], [77].

The PSD simulation results of the EFM code which is used in the current CD system, the EFMplus code which is used in the current DVD system, and other promising EFM-like codes show that cutoff frequency cannot accurately evaluate suppression of low-frequency components in the frequency band of current interest [10], [18]-[20], [31], [77]. It is desirable to develop new performance metrics to judge this performance. Measurement of PSD at the normalized frequency 10^{-3} [75], and subsequently at 10^{-4} [10], has been proposed as a metric to compare the spectrum performance of alternative codes. However, this new metric only

evaluates spectral content at individual specific frequencies, and as indicated in [18], the values of PSD at these frequencies can only be obtained by computer simulations. Although dc^1-free codes are currently widely used in practice, higher-order spectral-null codes will likely be required in future systems.

A new analytic method to assess low-frequency characteristics of codes in the frequency range of interest will be introduced in the next section. In contrast to expression (3.47) which reveals one important characteristic of dc^1-free sequences, the width of the spectral null, another important low-frequency property of dc^K-free sequences $(K \geq 1)$, the depth of the spectral null, will be explored in the next section. Based on this property, a new performance metric suitable for evaluation of arbitrarily high-order spectral-null codes is proposed. The significance of the introduction of the new performance metric is that it can not only be used in performance evaluation of constructed codes but can also be employed for design of good dc^K-free codes.

3.3 Performance Metric for Arbitrarily High-Order Spectral-Null Codes — Low-Frequency Spectrum-Weight (LFSW)

The main purpose of high-order spectral-null codes is to realize large rejection of low-frequency spectral components around zero frequency [22], [23]. Recall the definition and the characteristics of a code with a Kth-order spectral null at dc as outlined in Chapter 2 of this thesis.

In order to evaluate a coding scheme for spectral-null codes, all aspects of the code described in Section 1.2 need to be considered including coding complexity, code rate, power spectrum performance, and error propagation in decoding. It is usually desired that the spectral-null code be implemented in a simple manner, have a high information rate approaching rate capacity, have a large rejection of spectral components at low frequencies, and have limited error extension. In this section, the PSD performance of HOSN codes is addressed. Evaluation of the PSD of spectral-null codes allows comparison of their abilities to suppress spectral components at low frequencies. Rate-efficient HOSN codes and implementation of those codes will be discussed in the subsequent chapters.

Sum-variance overviewed in Section 3.2 is a well-known metric for assessing the performance of first-order spectral-null codes. However, as shown below in Section 3.3.1, while sum-variance metric is a valid metric for comparing the performance of first-order spectral-null codes, it is unsuitable for comparing the spectral compression of higher-order spectral-null codes because it is related to the width of the spectral null rather than taking into account the

depth of the spectral notch at low frequencies which is currently of significant interest [10], [18], [20], [31], [75]. This has motivated the development of a new metric indicative of the depth of the spectral notch at low frequencies for arbitrarily high-order spectral-null codes.

In Section 3.3.1, low-frequency characteristics of arbitrarily high-order spectral-null codes are discussed. A new performance metric for evaluating the spectrum compression of those codes around zero frequency is introduced; this metric is called the low-frequency spectrum-weight (LFSW). This subsection begins with several examples that demonstrate that a metric associated with the cutoff frequency fails to accurately assess the suppression of the spectral content of dc^K-free codes $(K > 1)$ at low frequencies, shows that the asymptotic low-frequency spectral components of Kth-order spectral-null codes $(K \geq 1)$ are exclusively determined by the order K and the LFSW, and demonstrates that the LFSW of dc^K-free codes equals the zero-frequency value in the spectrum of the sequence of Kth-order RDS values.

The LFSW for symbol-by-symbol encoding is first derived in Section 3.1.1, and then it is extended to block HOSN codes in Section 3.3.2. The characterization of high-order RDS sequences corresponding to encoded symbol sequences is then considered. Sequences of high-order RDS values corresponding to block coded symbol sequences are shown to be cyclostationary with period equal to the word length N. A modification of the well-known method for spectral analysis of block codes [34] discussed in Section 3.1.2 can be employed to calculate the LFSW of block dc^K-free codes.

A closed-form expression for the LFSW of dc^K-free codes $(K \geq 1)$ constructed through state-independent encoding, Kth-order zero-disparity codes, is derived, and an enumeration method to calculate this LFSW is outlined. The maximal and minimal asymptotic low-frequency spectral components of high-order zero-disparity codes are considered. It is shown that the difference between the minimal and maximal asymptotic low-frequency spectral components of K-OZD codes with the same codeword length and code rate can be in excess of 10 dB. Closed-form expressions for LFSW of first-order zero-disparity codes, for LFSW of low-disparity dc^1-free codes, and for the asymptotic LSFW of maxentropic dc^1-free codes are also given. The product of redundancy and square root of the asymptotic LFSW of maxentropic dc^1-free codes is shown to be asymptotically constant.

3.3.1 Asymptotic Power Spectral Density of HOSN Codes and LFSW

In this subsection, examples are presented to demonstrate that the cutoff frequency is not a valid metric on which to base comparison of the suppression of low-frequency components of

dc^K-free codes $(K > 1)$. Then a new performance metric, LFSW, which is a valid metric for comparison of codes with an arbitrary order of spectral null at dc, is derived.

A. Motivation

Consider the power spectra of 1/2 1-OZD (bi-phase), 1/4 2-OZD, and 1/8 3-OZD codes given in Figure 3.11. Note that when the width of spectral null is quantified by the cutoff frequency, it can be observed that the lower the order of spectral null, the wider the spectral null. In contrast, it is also clearly seen that at low frequencies, the higher the order of spectral null, the greater the rejection of spectral components. This phenomenon has also been observed in [22] for 1- and 2- OZD codes. In Figure 3.12, the power spectra of a 2/4 dc^2-free code constructed with three states [26] and the 3/8 2-OZD code are given. This figure illustrates that even when codes (with $K > 1$) have the same order of spectral null, it is possible to make incorrect decisions regarding the suppression of spectral components near dc when evaluation is based on the width (cutoff frequency) of the spectral null as opposed to the depth of the spectral null at very low frequencies. These figures demonstrate that the cutoff frequency cannot be generalized to be a consistently valid measure of spectrum rejection of low-frequency components for dc^K-free codes $(K > 1)$.
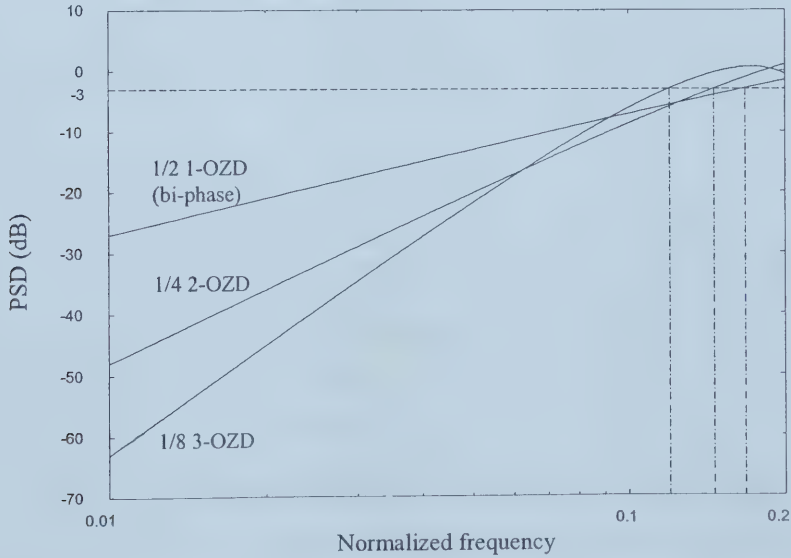
Figure 3.11 Comparison of cutoff frequencies of 1/2 1-OZD, 1/4 2-OZD, and 1/8 3-OZD codes.
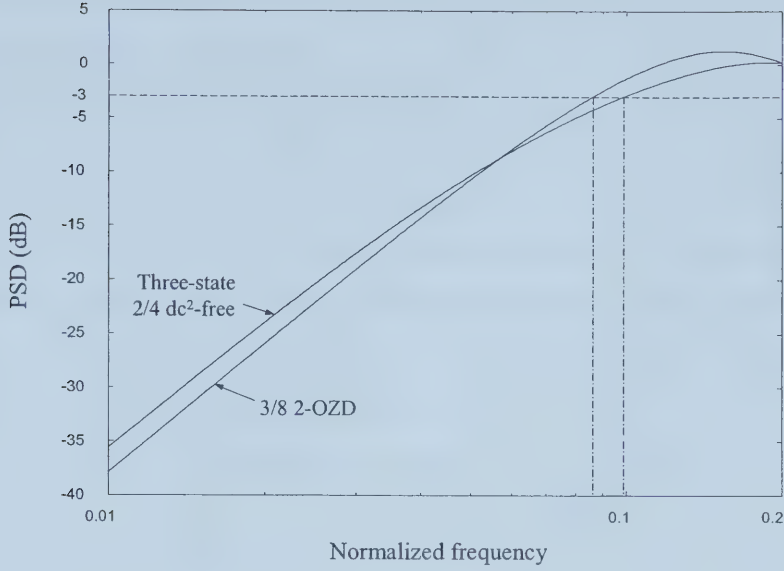
Figure 3.12 Comparison of cutoff frequencies of a three-state
2/4 dc²-free code and a 3/8 2-OZD code.

In addition, it is desirable to know the suppressed spectrum values at relatively low frequencies, e.g., at the normalized frequency 10^{-3} [18], [75] or 10^{-4} [10]. Although evaluation of the spectrum at specific frequencies increases the knowledge of the characteristics of the spectral null, this measure by definition only offers spectrum information at individual frequencies for HOSN codes and computer simulation is needed to complete the measure. Furthermore, this method can only be used for analysis of dc^K-free codes but not for synthesis of dc^K-free codes.

B. Low-Frequency Spectrum-Weight

As indicated in Section 2.2.2, a dc^K-free encoder can be modeled as a finite-state sequential machine (FSSM) [34]. Assuming that the input of the FSSM is a stationary and memoryless source sequence, the FSSM can be modelled as a stationary Markov chain [43]. Assume that it is also irreducible and aperiodic [46], an assumption that is valid for all codes of practical interest. For dc^K-constrained symbol-by-symbol encoding, let the state at instant l ($l \geq 1$), s_l ($s_l \in \mathcal{S}$, $\mathcal{S}$ is the set of encoder states), be represented by a vector with K finite values of $\text{RDS}^{(r)}$ ($r = 1, \cdots, K$), i.e., $s_l = [\text{RDS}_{l-1}^{(1)}, \cdots, \text{RDS}_{l-1}^{(K)}]$; these values are assumed to be finite due to the bounds on $\text{RDS}^{(r)}$ ($r = 1, \cdots, K$) [25], [26]. Let the encoded symbol sequence $\{x\} = \{x_1, x_2, \cdots, x_l, \cdots\}$ be composed of symbols from a finite alphabet $\mathcal{I}$. Consider the case

when the coded symbols are a memoryless state function of the state values, that is $x_l = f(s_l)$. It is well-known that the stochastic process $\{x\}$ is stationary [46].

Power spectra of stationary stochastic processes generally contain both continuous components and discrete components [34]. In the remainder of this thesis, only the continuous spectral component is considered. Let this continuous component of the power spectrum of the sequence $\{x\}$ be denoted as $H_x(\omega)$. Since the time sequence takes only real values, $H_x(\omega)$ is an even and non-negative function of ω. Assume that for dc^K-free coded sequences, $H_x(\omega)$ is $(2K+1)$-times differentiable in a neighborhood around dc. In this region, $H_x(\omega)$ near zero frequency can be expressed by the Taylor's formula [78, pp. 535-536]:

$$H_x(\omega) = H_x(0) + \frac{1}{2!}H_x^{(2)}(0)\omega^2 + \frac{1}{4!}H_x^{(4)}(0)\omega^4 + \cdots + \frac{1}{(2K)!}H_x^{(2K)}(0)\omega^{2K}$$

$$+ \frac{1}{(2K+1)!}H_x^{(2K+1)}(\omega^*)\omega^{2K+1}, \tag{3.56}$$

where $H_x(0), H_x^{(2)}(0), \cdots, H_x^{(2K)}(0)$ are constant and independent of ω, and $|\omega - \omega^*| \le |\omega|$ for specific codes. From the definition of dc^K-free codes [27], $H_x^{(k)}(0) = 0, k = 0, 1, \cdots, 2K-1$.[6] When ω approaches zero, the PSD of dc^K-free codes can be approximated as:

$$H_x(\omega) \approx \frac{1}{(2K)!}H_x^{(2K)}(0)\omega^{2K}. \tag{3.57}$$

Expression (3.57) demonstrates that:

- low-frequency spectral components of dc^K-free codes decrease asymptotically with decreasing frequency, with a slope of $20K$ dB/decade on a logarithmic scale. This has been observed in [26] for dc^2-free codes;

- given the order K, the smaller the value of $H_x^{(2K)}(0)$, the greater the suppression of low-frequency spectral components by the dc^K-free code;

- given different orders of spectral null K_1 and K_2, the approximate cross-over of spectra of the dc^{K_1}-free codes and dc^{K_2}-free codes is uniquely determined by the values of $H_x^{(2K_1)}(0)$ and $H_x^{(2K_2)}(0)$.

---

[6] For convenience, $H_x^{(0)}(0)$ is written as $H_x(0)$.

Because $H_x^{(2K)}(0)$ reflects the depth of the spectral null at low frequencies and its value depends on the coding rules, this value is proposed to be used as a means to compare the quality of Kth-order spectral-null codes. The following simple examples demonstrate the validity of approximation (3.57). Equiprobable and independent source bits are assumed in these and all following examples.

Example 3.2: Closed-form expressions for power spectra of 1/2 1-OZD, 1/4 2-OZD, and 1/8 3-OZD codes have been given in (3.30), (3.31), and (3.32) respectively. By taking the second, fourth, and sixth derivative to the expressions (3.30), (3.31), and (3.32) respectively and by letting the frequency be zero, it can be shown that $H_{x1}^{(2)}(0)=1$, $H_{x2}^{(4)}(0)=24$, and $H_{x3}^{(6)}(0)=5760$. ◇

By using these values of $H_{x1}^{(2)}(0)$, $H_{x2}^{(4)}(0)$, and $H_{x3}^{(6)}(0)$ in (3.57), the approximate low-frequency spectra for the 1/2 1-OZD (bi-phase), 1/4 2-OZD, and 1/8 3-OZD codes respectively can be obtained. Figure 3.13 compares the exact (solid curves) and approximate (dashed lines) power spectra of those codes. At normalized frequencies lower than 0.01, the percentage difference between the exact and approximate spectrum values (in dB) of the 1/2 1-OZD, 1/4 2-OZD, and 1/8 3-OZD codes are less than $5.3\times10^{-3}\%$, $1.5\times10^{-2}\%$, and $4.8\times10^{-2}\%$ respectively.
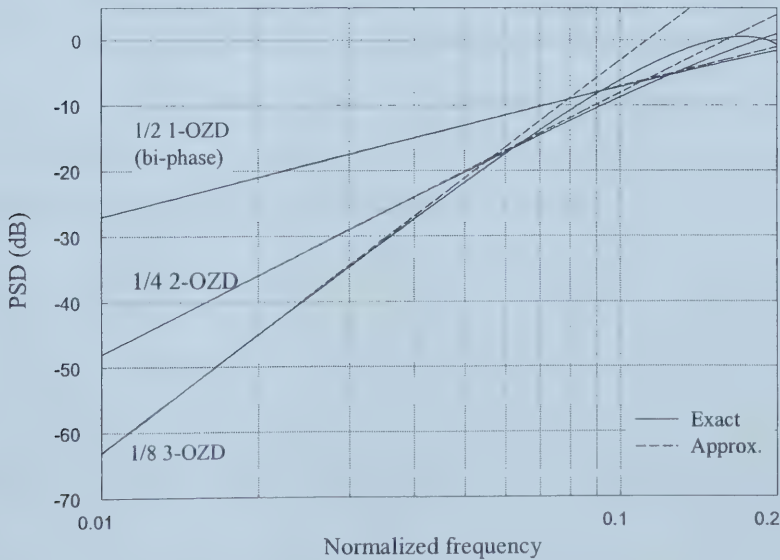
Figure 3.13 Comparison of exact and approximate power spectra of
1/2 1-OZD, 1/4 2-OZD, and 1/8 3-OZD codes.

Although $H_x^{(2K)}(0)$ essentially characterizes dc^K-free codes at low frequencies, the need to determine an analytic expression for $H_x(\omega)$, and then calculate relatively high derivatives of $H_x(\omega)$ appears to make analysis of $H_x^{(2K)}(0)$ impractical.[7] The challenge is to find a method to conveniently evaluate the value of $H_x^{(2K)}(0)$.

For the sequence $\{x\}$ with a Kth-order spectral null at dc, $\text{RDS}_l^{(K)}$ ($l \geq 1$), the Kth-order RDS at instant l, is related to the state $s_l = [\text{RDS}_{l-1}^{(1)}, \cdots, \text{RDS}_{l-1}^{(K)}]$ and the symbol x_l through Equation (2.7). $\text{RDS}_l^{(K)}$ is a memoryless function of both s_l and x_l, i.e. $\text{RDS}_l^{(K)} = \varphi(s_l, x_l)$. Since it is assumeed that states are defined such that $x_l = f(s_l)$, $\text{RDS}_l^{(K)}$ is a memoryless function of only s_l, that is $\text{RDS}^{(K)}(l) = \Phi(s_l)$. Characterization of a memoryless function of a stationary Markov chain was extensively developed in [46]. Based on their analysis, it can be shown that the stochastic process $\{\mathbf{RDS}^{(K)}\}$, the sequence of $\text{RDS}^{(K)}$ values corresponding to the sequence $\{x\}$, is stationary.

Let $R_x(k)$ and $R_{\text{RDS}^{(K)}}(k)$ be the autocorrelation functions of the sequences $\{x\}$ and $\{\mathbf{RDS}^{(K)}\}$ respectively, and let $H_x(\omega)$ and $H_{\text{RDS}^{(K)}}(\omega)$ be the corresponding spectra. From the recursive relationship in (2.7), there exists:

$$\text{RDS}_l^{(K-1)} = \text{RDS}_l^{(K)} - \text{RDS}_{l-1}^{(K)} . \tag{3.58}$$

It can then be shown that:

$$R_{\text{RDS}^{(K-1)}}(k) = 2R_{\text{RDS}^{(K)}}(k) - R_{\text{RDS}^{(K)}}(k+1) - R_{\text{RDS}^{(K)}}(k-1) \tag{3.59}$$

where $R_{\text{RDS}^{(K)}}(k) = E\{\text{RDS}_l^{(K)} \text{RDS}_{l+k}^{(K)}\}$ and $E\{\ \}$ denotes expectation. Because the autocorrelation function and power spectrum density function are a Fourier transform pair, a similar recursive relation in the frequency domain can be obtained:

$$H_{\text{RDS}^{(K-1)}}(\omega) = 2H_{\text{RDS}^{(K)}}(\omega)(1 - \cos\omega) . \tag{3.60}$$

Since $\text{RDS}_l^{(0)} = x_l$, there is:

$$H_x(\omega) = H_{\text{RDS}^{(0)}}(\omega) = 2^{2K} H_{\text{RDS}^{(K)}}(\omega)(\sin(\omega/2))^{2K} . \tag{3.61}$$

Consequently, Theorem 3.4 follows.

Theorem 3.4: If $H_x^{(i)}(\omega)$ ($i = 0, \cdots, 2K$) and $H_{\text{RDS}^{(K)}}(\omega)$ are continuous at dc, then for dc^K-free codes:

$$H_x^{(2K)}(0) = H_x^{(2K)}(\omega)\Big|_{\omega=0} = (2K)! H_{\text{RDS}^{(K)}}(0) . \tag{3.62}$$

---

[7] Very few dc^K-free codes have simple expressions for PSD.

Proof of Theorem 3.4 is performed by evaluating the $2K$th derivative to both sides of (3.61) and then by letting $\omega = 0$. Combining (3.57) and (3.62) yields the following approximation for $H_x(\omega)$ at low frequencies:

$$H_x(\omega) \approx H_{RDS^{(K)}}(0)\omega^{2K}. \tag{3.63}$$

Denote $H_{RDS^{(K)}}(0)$ to be the *low-frequency spectrum-weight* (LFSW) of dc^K-free codes. It is clear from (3.63) that the magnitude of the low-frequency components of high-order spectral-null codes can be compared using their values of $H_{RDS^{(K)}}(0)$ suggesting LFSW as a performance metric for evaluation of low-frequency suppression of arbitrarily high-order spectral-null codes.

Corollary 3.1: Let f_c be the cross-over frequency of spectra of a dc^{K_1}-free code and a dc^{K_2}-free code $(K_1 \neq K_2)$. Then:

$$f_c \approx (1/2\pi)\left[\frac{H_{RDS^{(K_1)}}(0)}{H_{RDS^{(K_2)}}(0)}\right]^{\frac{1}{2(K_2-K_1)}}. \tag{3.64}$$

Proof of Corollary 3.1 is straightforward using (3.63). Approximation (3.64) can be used to estimate that at normalized frequencies lower than the cross-over frequency, the code with the higher-order spectral null at dc has superior suppression of spectral components.

It is now shown that the LFSW is independent of the initial value of $RDS^{(r)}$, $r = 1, \cdots, K$. Assume that the sequence $\{RDS^{(K)}\}$ is asymptotically uncorrelated [34], an assumption which follows from the Markov chain being stationary. $H_{RDS^{(K)}}(0)$ can be expressed as:

$$H_{RDS^{(K)}}(0) = \sum_{k=-\infty}^{\infty}\left[R_{RDS^{(K)}}(k) - m_{RDS^{(K)}}^2\right], \tag{3.65}$$

where $m_{RDS^{(K)}}$ is the mean of $\{RDS^{(K)}\}$, i.e. $m_{RDS^{(K)}} = E\{RDS_l^{(K)}\}$.

Let C denote the set of complex numbers. For sequences with a Kth-order spectral null at dc, there exist the following state functions $\psi_r (r=1,\cdots,K)$: $\mathcal{S} \rightarrow C$ [25, Theorem 2 (a)]:

$$x_l = \psi_1(s_{l+1}) - \psi_1(s_l)$$

$$\psi_{r-1}(s_{l+1}) = \psi_r(s_{l+1}) - \psi_r(s_l), \text{ for } 2 \leq r \leq K \text{ and } l \geq 1. \tag{3.66}$$

Equivalently, the Kth-order RDS at the instant l can be expressed by [25, Theorem 2 (b)]:

$$RDS^{(1)}(l) = \psi_1(s_{l+1}) - \psi_1(s_l)$$

$$RDS^{(r)}(l) = \psi_r(s_{l+1}) - \psi_r(s_l) - \sum_{q=1}^{r-1}\psi_{r-q}(s_l)\binom{l-1+q}{q}, \quad 2 \leq r \leq K \tag{3.67}$$

where s_1 is the initial state. Given the initial state s_1 with initial values $\text{RDS}^{(r)}(0)$, $r = 1, \cdots, K$, the value of $\psi_r(s_1)$ is deterministic. Denote:

$$\Psi_K(s_1) = \psi_K(s_1) + \sum_{q=1}^{K-1} \psi_{K-q}(s_1) \binom{l-1+q}{q}. \tag{3.68}$$

The value of $\Psi_K(s_1)$ is also deterministic. Based on the theory in [46], the stochastic process $\psi_K(s_{l+1})$, which is a memoryless function of the states, is wide-sense stationary. From (3.65) there is:

$$H_{\text{RDS}^{(K)}}(0) = \sum_{k=-\infty}^{\infty} \left[E\{\text{RDS}_l^{(K)}\text{RDS}_{l+k}^{(K)}\} - (E\{\text{RDS}_l^{(K)}\})^2 \right]. \tag{3.69}$$

Combining (3.67) and (3.68) with (3.69) yields:

$$H_{RDS^{(K)}}(0) = \left[E\{\text{RDS}_l^{(K)}\text{RDS}_{l+k}^{(K)}\} - (E\{\text{RDS}_l^{(K)}\})^2 \right]$$

$$= \sum_{k=-\infty}^{\infty} \left[E\{[\psi_K(s_{l+1}) - \Psi_k(s_1)][\psi_K(s_{l+1+k}) - \Psi_k(s_1)]\} - (E\{\psi_K(s_{l+1}) - \Psi_k(s_1)\})^2 \right]$$

$$= E\{\psi_K(s_{l+1})\psi_K(s_{l+1+k})\} - (E\{\psi_K(s_{l+1})\})^2.$$

Therefore the LFSW of dc^K-free codes is independent of the initial values of $\text{RDS}^{(r)}$ $(r = 1, \cdots, K)$.

Low-frequency characteristics of symbol-by-symbol coded sequences have been discussed above. However, as indicated in Section 2.2.2, to realize limited error propagation during decoding, block decoders and sliding block decoders are usually employed in dc^K-free coding. In practice, block coding is employed to provide mapping between the source sequence and the coded sequence in dc^K-free codes. In the subsequent subsections, LFSW of block HOSN codes will be discussed.

3.3.2   LFSW of Codes Generated through State-Dependent Encoding

Consider dc^K-free block coded sequences where the encoded symbols are from a finite alphabet. Let the encoded symbol sequence $\{x\}$ be the concatenation of length-N encoded codewords $\{\underline{x}\} = \{\underline{x}_1, \underline{x}_2, \cdots, \underline{x}_n, \cdots\}$, where the nth codeword is $\underline{x}_n = [x_{n,1}, x_{n,2}, \cdots, x_{n,N}]$, $x_{n,j} \in \mathcal{I}$, $j = 1, \cdots, N$. Denote $x_{n,j} = x_l$, where $l = (n-1)N + j$. Also let $\underline{\text{RDS}}_n^{(r)} = [\text{RDS}_{n,1}^{(r)}, \text{RDS}_{n,2}^{(r)}, \cdots, \text{RDS}_{n,N}^{(r)}]$ be a vector of rth-order RDS values $(r = 1, \cdots, K)$ corresponding to symbols in the codeword $\underline{x}_n$, and let $\{\underline{\text{RDS}}^{(r)}\}$ be the sequence of $\underline{\text{RDS}}_n^{(r)}$ words. Denote $\text{RDS}_{n,j}^{(K)} = \text{RDS}_{(n-1)N+j}^{(K)} = \text{RDS}_l^{(K)}$.

Let each state s_n, $s_n \in \mathcal{S} = \{\sigma_1, \cdots, \sigma_S\}$ (S is the number of states), of the finite-state Markov chain represent K finite values of $\mathrm{RDS}^{(1)}, \mathrm{RDS}^{(2)}, \ldots, \mathrm{RDS}^{(K)}$ at the end of the $(n-1)$th codeword. Therefore, $\mathrm{RDS}^{(r)}$ denotes values of rth-order RDS prior to transmission of the nth codeword, i.e. $\mathrm{RDS}^{(r)}_{n-1,N} = \mathrm{RDS}^{(r)}_{n,0}$, $r = 1, \cdots, K$. The vector of Kth-order RDS values, $\underline{\mathrm{RDS}}^{(K)}_n = [\mathrm{RDS}^{(K)}_{n,1}, \mathrm{RDS}^{(K)}_{n,2}, \cdots, \mathrm{RDS}^{(K)}_{n,N}]$, is related to the symbols in the nth codeword $\underline{x}_n$ and the initial values of the rth-order RDS for $\underline{x}_n$, $\mathrm{RDS}^{(r)}_{n,0}$, through Equation (2.7). As a result, the values of Kth-order RDS in $\underline{\mathrm{RDS}}^{(K)}_n$ depend only on $\underline{x}_n$ and the state s_n.

Let the outputs of the encoder, $\{\underline{x}_n\}$ and $\{\underline{\mathrm{RDS}}^{(K)}_n\}$, be vector-valued stochastic processes. Similar to the case of symbol-by-symbol encoding, the processes $\{\underline{x}\}$ and $\{\underline{\mathbf{RDS}}^{(K)}\}$ are wide-sense stationary implying that $\{x\}$ and $\{\mathbf{RDS}^{(K)}\}$ are wide-sense cyclostationary with period equal to the codeword length N [34].

The calculation of spectra of cyclostationary processes has been extensively developed [34], [66]. Based on the approach for evaluation of autocorrelation outlined in [34], which was also discussed in Section 3.1.2 of this thesis, $H_{\mathrm{RDS}^{(K)}}(0)$ can be obtained with the modification that the output of the finite-state sequential machine that is used in the calculation is the sequence of Kth-order RDS values rather than the sequence of symbols.

Assume that the input of the FSSM is a binary word sequence $\{\underline{y}\}$. Let the $U = 2^M$ source words with length M be represented by M-dimensional row vectors $\underline{y}(u) = [y_1(u), \cdots, y_M(u)]$ ($u = 0, 1, \cdots, U-1$), and let the source word at the nth coding interval be $\underline{y}_n = [y_{n,1}, y_{n,2}, \cdots, y_{n,M}]$ where $\underline{y}_n \in \{\underline{y}(u)\}$. Similar to the descriptions in Section 2.2.2, state transitions can be described as $s_{n+1} = g(s_n, \underline{y}_n)$ where g is the state transition function of the FSSM, and s_n and s_{n+1} are the present and next states respectively. As noted above, for calculation of $H_{\mathrm{RDS}^{(K)}}(\omega)$, consider the output of the FSSM to be the sequence of Kth-order RDS values rather than a sequence of symbols, retain the representations of q_u, $\mathbf{D}$, E_u, π, $\mathbf{Q}$, Π_∞, and U as defined in Section 3.1.2, and redefine Γ_u in this subsection as the $S \times N$ dimensional matrix in which the ith row ($i = 1, \cdots, S$) of Γ_u is the output of Kth-order RDS values corresponding to the output word and the ith state when the input to the encoder is the uth

source word. Denote $m_{\text{RDS}^{(K)}}$ as the length-N mean vector of the word sequence of Kth-order RDS values. Since the value of LFSW is the value of $H_{\text{RDS}^{(K)}}(\omega)$ evaluated at zero frequency, calculation of the complete spectrum of the sequence of Kth-order RDS values is not needed. By following the calculation of spectra in described in Section 2.2.2, an $N \times N$ matrix $H_{\text{RDS}^{(K)}}(0)$ can be described as $H_{\text{RDS}^{(K)}}(0) = Y(1) + Y'(1)$, with

$$Y(1) = (1/2)(C_0 - m'_{\text{RDS}^{(K)}} m_{\text{RDS}^{(K)}}) + C_1 B(1) C_2$$

where the prime denotes transposition and C_0, C_1, C_2 and $B(1)$ [8] are as defined in Theorem 3.1, but with our new definition of Γ_u. Similar to (3.17), the average continuous part of the spectrum of Kth-order RDS values is defined as:

$$H_{\text{RDS}^{(K)}}(0) = \frac{1}{N} w' H_{\text{RDS}^{(K)}}(0) w,$$

where w is the length-N all-one column vector.

Example 3.3: Consider calculation of LFSW of the 2/4 four-state dc^2-free code constructed in [22]. The code table is given as below.

Table 3.1 Output codewords and state transitions of the 2/4 dc^2-free code (4-state) [22]

Source word	$S_n = \sigma_1$		$S_n = \sigma_2$		$S_n = \sigma_3$		$S_n = \sigma_4$	
	Codeword	S_{n+1}	Codeword	S_{n+1}	Codeword	S_{n+1}	Codeword	S_{n+1}
00	+ − − +	σ_1	+ − − +	σ_2	+ − − +	σ_3	+ − − +	σ_4
01	− + + −	σ_1	− + + −	σ_2	− + + −	σ_3	− + + −	σ_4
10	+ − + −	σ_2	− + − +	σ_1	− + − +	σ_2	+ − + −	σ_1
11	+ + − −	σ_3	− − + +	σ_4	− − + +	σ_1	+ + − −	σ_2

There are total $2^2 = 4$ source words and four states (σ_s, $s = 1, \cdots, 4$) which are represented by (RDS$^{(1)}$, RDS$^{(2)}$). Let E_u be the 4×4 state transition matrix of the uth source word ($u = 0, \cdots, 3$). From Table 3.1, it can be found that:

$$E_0 = E_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}; \quad E_2 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}; \quad E_3 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$

---

[8] Note that $z = \exp(j2\pi fT)$ in the corresponding expressions in [34] reduces to $z = 1$ for calculation of LFSW.

It is straightforward from (3.23) to show that the transition probability matrix Q is:

$$Q = \begin{bmatrix} 0.5 & 0.25 & 0.25 & 0 \\ 0.25 & 0.5 & 0 & 0.25 \\ 0.25 & 0.25 & 0.5 & 0 \\ 0.25 & 0.25 & 0 & 0.5 \end{bmatrix}.$$

Furthermore, the steady state probability vector π is:

$$\pi = [0.3333 \ 0.3333 \ 0.1667 \ 0.1667].$$

Let $Q_\infty = w\pi$, let U be the identity matrix, and let D be the diagonal matrix $D = \text{diag}\{\pi(1), \cdots, \pi(4)\}$. Let Γ_u be the 4×4 output matrices of the $RDS^{(2)}$ values.

$$\Gamma_0 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 3 & 3 & 2 & 2 \\ 5 & 5 & 4 & 4 \\ -1 & -1 & -2 & -2 \end{bmatrix}; \Gamma_1 = \begin{bmatrix} -1 & -1 & 0 & 0 \\ 1 & 1 & 2 & 2 \\ 3 & 3 & 4 & 4 \\ -3 & -3 & -2 & -2 \end{bmatrix}; \Gamma_2 = \begin{bmatrix} 1 & 1 & 2 & 2 \\ 1 & 1 & 0 & 0 \\ 3 & 3 & 2 & 2 \\ -1 & -1 & 0 & 0 \end{bmatrix}; \Gamma_3 = \begin{bmatrix} 1 & 3 & 4 & 4 \\ 1 & -1 & -2 & -2 \\ 3 & 1 & 0 & 0 \\ -1 & 1 & 2 & 2 \end{bmatrix};$$

The mean vector of Kth-order RDS values is $m_{RDS^{(2)}} = \sum_{u=0}^{3} q_u \pi \Gamma_u$ where q_u is the probability of the uth source word. In this example, $q_u = 0.25$, and $m_{RDS^{(2)}} = [1 \ 1 \ 1 \ 1]$.

Using these definitions and the approach outlined above, the LFSW of the 2/4 four-state dc^2-free code (the continuous part of the spectrum of the sequence of $RDS^{(2)}$ values at zero frequency) can be obtained:

$$H_{RDS^{(2)}}(0) = 30.2778.$$ ◊

Example 3.4: Similarly, consider calculation of LFSW of the three-state 2/4 dc^2-free code designed in [26]. The code table of this new approach is shown in Table 3.2 below.

Table 3.2 Output codewords and state transition of the 2/4 dc^2-free code (3-state) [26]

Source word	$S_n = \sigma_1$		$S_n = \sigma_2$		$S_n = \sigma_3$	
	Codeword	S_{n+1}	Codeword	S_{n+1}	Codeword	S_{n+1}
00	$-++-$	σ_1	$-++-$	σ_2	$-++-$	σ_3
01	$+--+$	σ_1	$+--+$	σ_2	$+--+$	σ_3
10	$+-+-$	σ_2	$+-+-$	σ_3	$--++$	σ_1
11	$++--$	σ_3	$-+-+$	σ_1	$-+-+$	σ_2

Note that compared to the code in Example 3.3, although the code rate of this code is the same, with rearrangement of the mapping between source words and codewords, the number of states

has been reduced from four to three. In the same manner as in Example 3.3, the value of LFSW of the three-state 2/4 dc^2-free code is found to be:

$$H_{RDS^{(2)}}(0) = 18.2778 \,.$$ ◊

Based on the values of LFSW for the 2/4 dc^2-free codes in Examples 3.3 and 3.4, and relationship (3.63), the approximate low-frequency spectra of these two codes are plotted in Figure 3.14. For purpose of comparison, the exact spectra of these codes shown in [26] are also given. Below the normalized frequency 0.01, the difference between the exact and approximate spectral component values (in dB) of the 2/4 three-state and four-state dc^2-free codes is less than 0.9% . The increased suppression by about 2 dB in the low-frequency spectral components of the three-state dc^2-free code observed in [26] was heuristically explained by the reduced RDS^2 variation. Here, this observation can be well-explained by the decrease by about 2 dB of the LFSW value of this code.
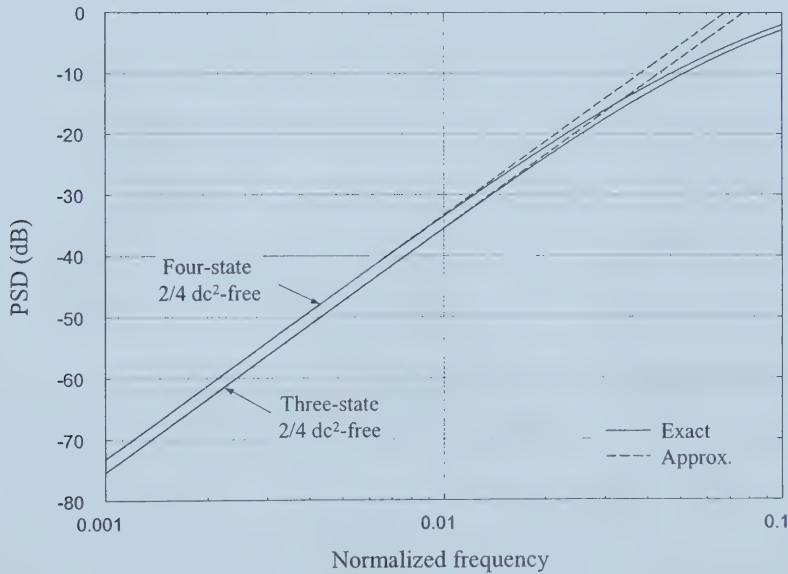
Figure 3.14 Comparison of exact and approximate power spectra of
the three-state and four-state dc^2-free codes.

More complicated coding schemes such as multimode coding will be discussed in Chapter 4 and Chapter 5 of this thesis. Recently, dc^1-free multimode codes and their performance have proven to be of interest [1, ch.13], [10], [21], [31]. Guided scrambling (GS) [79] is one multimode coding technique that will be addressed in Chapter 4. This thesis considers efficient dc^K-free GS codes ($K \geq 1$) with large rejection of low-frequency components. Although analysis

and design of these efficient codes are discussed later, to demonstrate the suitability of LFSW for complicated cases, in Figure 3.15, exact low-frequency spectra and approximate spectra of GS 4/8 dc²-free, GS 10/16 dc²-free, and GS 17/24 dc²-free codes are presented. For all three of these GS codes, the selection criterion is to select the encoded word with the minimal sum of squared $RDS^{(2)}$ values from the selection set over the duration of that word. In the legend of this figure, d denotes the coefficients of the scrambling polynomial. Below the normalized frequency 0.01, the percentage difference between the exact and approximate spectral component values (in dB) of all three codes is less than 1.3% .
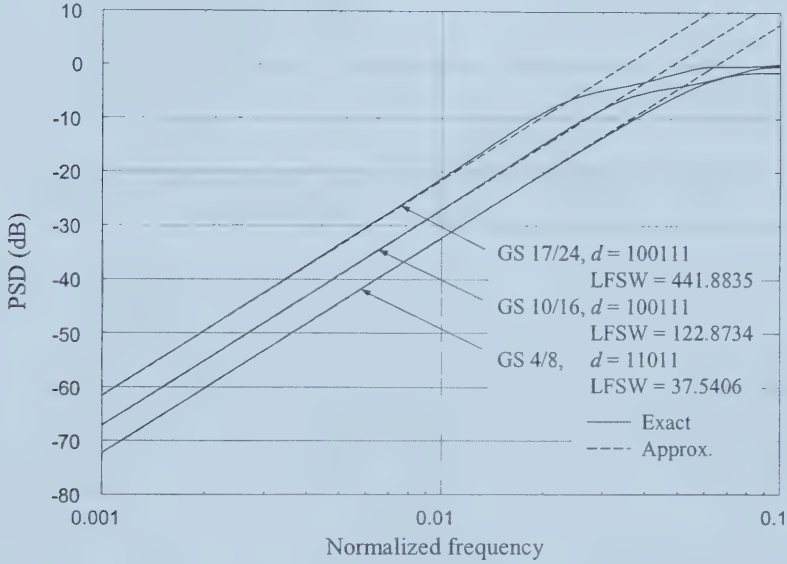
Figure 3.15 Comparison of exact and approximate power spectra of GS 4/8, GS 10/16, and GS 17/24 dc²-free codes.

As described above, the application of the new LFSW metric is not limited to coded symbol sequences from the bipolar alphabet but from any finite alphabet. Figure 3.16 compares the exact and approximate power spectra of two ternary codes, the well-known AMI code [40] (an example of a 1B1T dc¹-free code) and the 3B/3T dc²-free code constructed in [26]. The power spectrum of the AMI code has the closed-form expression $H(\omega) = (1 - \cos\omega)/2$ [40], [62].

According to Figure 3.16, below the normalized frequency 0.01, the percentage difference between the exact and approximate spectral component values (in dB) of these codes is less than 0.1% . From (3.64), the estimated cross-over frequency of power spectra of these codes can be obtained as $f_c \approx 2.336 \times 10^{-2}$ while the actual cross-over frequency in this case is 2.396×10^{-2} .
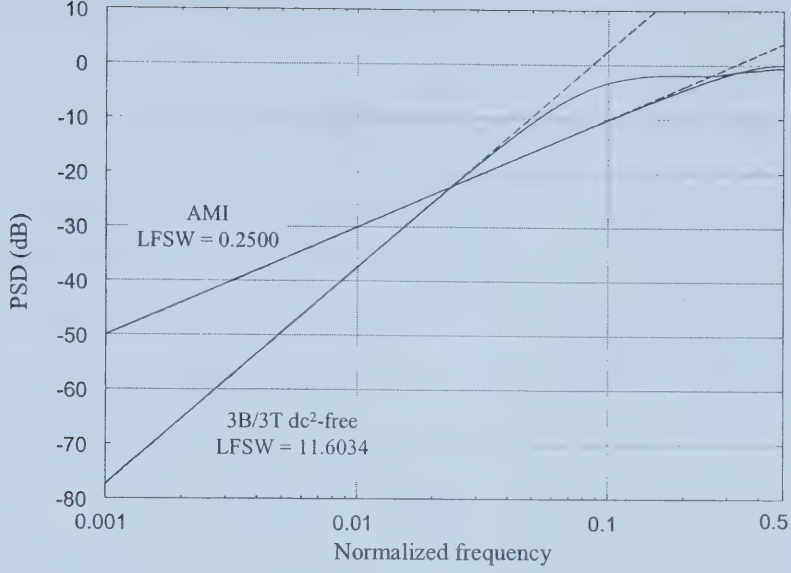
Figure 3.16 Comparison of exact and approximate power spectra of
AMI and 3B/3T dc²-free codes.

3.3.3   LFSW of High-Order Zero-Disparity Codes

In a state-independent encoder, there exists only one principal state [26], and therefore, the outputs of the FSSM are independent of previous inputs or outputs. This condition makes it possible to simplify the calculation of LFSW. One class of state-independent encoders includes the encoders of Kth-order zero-disparity codes $(K \geq 1)$.

Consider a Kth-order zero-disparity code constructed through free concatenation of dc^K-free codewords of length N. Assume that codewords are equiprobable and that the complement of each codeword is another codeword. Let the set of dc^K-free words be $\mathcal{A}(N, K)$, let the number of codewords be A, let $\underline{x}_a = [x_{a,1}, x_{a,2}, \cdots, x_{a,N}]$ be an element in the set $\mathcal{A}(N, K)$, and let $\underline{RDS}_a^{(K)} = [RDS_{a,1}^{(K)}, RDS_{a,2}^{(K)}, \cdots, RDS_{a,N}^{(K)}]$ be the vector of $RDS^{(K)}$ values related to $\underline{x}_a$ through (2.7) where the initial value of the rth-order RDS of the codeword $\underline{x}_a$, $RDS_{a,0}^{(r)} = 0$ $(1 \leq r \leq K)$, is assumed to be zero. The rth-order moment of $\underline{x}_a$ $(r \geq 0)$, $u_a^{(r)}$, is defined in Section 2.3.1 as

$$u^{(r)} = \sum_{j=1}^{N} j^r x_j.$$ Recalling (2.30), for a dc^K-free word $\underline{x}_a$, there exists:

$$u_a^{(r)} = 0, \quad \text{for } r = 0, 1, \cdots, K-1.$$

Theorem 3.5: The LFSW of K-OZD codes is:

$$H_{\mathrm{RDS}^{(K)}}(0) = \frac{\sum_{a=0}^{A-1}\left(u_a^{(K)}\right)^2}{AN(K!)^2}.$$ (3.70)

Proof: This theorem is proved with two approaches.

Approach 1: Due to the assumption that $\mathrm{RDS}_{a,0}^{(r)} = 0$ ($1 \le r \le K$) and the set of codewords is composed of equiprobable and complementary codewords, the mean vector $\boldsymbol{m}_{\mathrm{RDS}^{(K)}}$ is the all-zero vector. Based on the analysis of LFSW in Section 3.3.2, it can be shown that for K-OZD codes:

$$H_{\mathrm{RDS}^{(K)}}(0) = \frac{1}{AN}\sum_{a=0}^{A-1}\left(\sum_{j=1}^{N}\mathrm{RDS}_{a,j}^{(K)}\right)^2.$$ (3.71)

Also let $\mathrm{RDS}_{a,0}^{(K+1)} = 0$. From (2.7), it is straightforward to find that:

$$\sum_{j=1}^{N}\mathrm{RDS}_{a,j}^{(K)} = \mathrm{RDS}_{a,N}^{(K+1)}.$$ (3.72)

The relationship of the Kth-order RDS value, $\mathrm{RDS}^{(K)}$, and the related rth-order moment ($r = 0, 1, \cdots, K-1$), $u^{(r)}$, of a sequence is presented in Lemmas 1 and 2 in [24] where the initial value of an arbitrarily high order RDS of the sequence is assumed to be zero. This relation is applied to individual dc^K-free words. According to the definition of dc^K-free words given in (2.30), it can be obtained that:

$$\mathrm{RDS}_{a,N}^{(K+1)} = \frac{(-1)^K u_a^{(K)}}{K!}.$$ (3.73)

By combining (3.72) and (3.73) with (3.71), the theorem follows.

Approach 2: Based on the assumption for equiprobable and complementary codewords outlined above, the power spectrum $H_x(\omega)$ of the state-independent encoded sequence can be expressed as [23]:

$$H_x(\omega) = \frac{1}{AN}\sum_{a=0}^{A-1}|X_a(\omega)|^2,$$ (3.74)

where $X_a(\omega)$ is the discrete Fourier transform of codeword $\underline{x}_a$:

$$X_a(\omega) = \sum_{j=1}^{N}x_{a,j}e^{-ij\omega}, \quad -\pi \le \omega \le \pi$$ (3.75)

Combining (3.75) with (3.74) yields:

$$H_x(\omega) = \frac{1}{AN} \sum_{a=0}^{A-1} \left[\left(\sum_{j=1}^{N} (x_{a,j}) \cos(\omega j) \right)^2 + \left(\sum_{j=1}^{N} (x_{a,j}) \sin(\omega j) \right)^2 \right]. \tag{3.76}$$

By taking $2K$ derivatives of both sides of (3.76), letting $\omega = 0$, and using (2.30), it can be shown that:

$$H_x^{(2K)}(0) = \frac{1}{AN} \binom{2K}{K} \sum_{a=0}^{A-1} \left(\sum_{j=1}^{N} j^K x_{a,j} \right)^2 = \frac{1}{AN} \binom{2K}{K} \sum_{a=0}^{A-1} \left(u_a^{(K)} \right)^2.$$

Combining this with (3.62) completes the proof. ☐

The simple codes considered in Example 3.2 are now revisited to verify the expression for LFSW of K-OZD codes ($1 \leq K \leq 3$).

Example 3.5: For the two codewords in the 1/2 1-OZD code, $A = 2$, $N = 2$, $K = 1$, $u_a^{(0)} = 0$ and $u_a^{(1)} = \pm 1$. From (3.70), $H_{\text{RDS}^{(1)}}(0) = 0.5$. For the two codewords in the 1/4 2-OZD code, $u_a^{(0)} = u_a^{(1)} = 0$ and $u_a^{(2)} = \pm 4$. From (3.70), $H_{\text{RDS}^{(2)}}(0) = 1$. For the two codewords in the 1/8 3-OZD code, $u_a^{(0)} = u_a^{(1)} = u_a^{(2)} = 0$ and $u_a^{(3)} = \pm 48$. Therefore $H_{\text{RDS}^{(3)}}(0) = 8$. Taking into account the factorial in (3.62), it can be found that the results for Example 3.5 are in agreement with the results previously found in Example 3.2. ◊

From Theorem 3.5, it is clear that the asymptotic low-frequency spectral components of K-OZD codes are exclusively determined by the order of the spectral-null K, the codeword length N, the number of dc^K-free codewords, and the sum of the squared Kth-order moments of the codewords. The average asymptotic low-frequency spectral components (or, equivalently, the average LFSW) of K-OZD codes can be evaluated by considering the full set of words $\mathcal{A}(N, K)$ through the assumption that all available dc^K-free words are used as codewords with equal probability. In this case, the number of codewords A equals A_c, the cardinality of the set $\mathcal{A}(N, K)$. Calculation of power spectra of K-OZD codes ($K \geq 1$) generated through full-set coding was discussed in Section 3.1.2 where power spectra of K-OZD codes ($1 \leq K \leq 4$) using full-set words were presented. Generally, full-set coding is nonconstructive and is used to simplify analysis. For a practical M/N K-OZD code, due to the one-to-one mapping between source words and codewords, the practical number of codewords is a power of two, i.e. $A = A_p = 2^M$ where $M = \lfloor \log_2 A_c \rfloor$. In some cases, it may also be of interest to know the performance limitation of an implemented K-OZD code with a given word length. From Theorem 3.5, it is straightforward to calculate the minimal and maximal LFSWs of a K-OZD

code by considering the required number of dc^K-free codewords with minimal and maximal sum of squared Kth-order moments respectively. Note that when $A_p = A_c$, the average, minimal, and maximal LFSWs are identical.

In the following, the minimal and maximal LFSWs as well as the corresponding minimal and maximal asymptotic low-frequency spectral components are evaluated for K-OZD codes.

Example 3.6: Consider the minimal and maximal asymptotic low-frequency spectral components of 3/16 3-OZD codes. The code book and the values of $u_a^{(3)}$ of the employed codewords are given in Table 3.3. Code I and code II are constructed with the maximal and minimal sums of squared third-order moment respectively. From Theorem 3.5, it is straightforward to show that for code I, $H_{RDS^{(3)}}(0) = 370$, whereas for code II, $H_{RDS^{(3)}}(0) = 45$. ◊

Table 3.3 Code book and $u_a^{(3)}$ of codewords of 3/16 3-OZD codes

Source word	Code I (maximal LFSW)		Code II (minimal LFSW)	
	Codeword	$u_a^{(3)}$	Codeword	$u_a^{(3)}$
000	− − + + + + − − + + − − − − + +	768	− + + − − + + − + − − + + − − +	192
001	− + − + + − + − + − + − − + − +	384	− + + − + − − + − + + − + − − +	96
010	− + − + + − + + − − − − − + + + −	240	− + + − + − − + + − − + − + + −	0
011	− + + + − − − − + + − + + − + −	-240	− + + + − − − − + + − + + − + −	-240
100	+ − − − + + + + − − + − − − + − +	240	+ − − − + + + + − − + − − − + − +	240
101	+ − + − − + − − + + + + + − − − +	-240	+ − − + − + + − − + + − + − − +	0
110	+ − + − − + − + − + − + + − + −	-384	+ − − + − + + − + − − + − + + −	-96
111	+ + − − − − + + − − + + + + − −	-768	+ − − + + − − + − + + − − + + −	-192

Figure 3.17 compares the exact spectra (solid curves) and approximate spectra (dashed lines) for the 3/16 3-OZD codes outlined in Table 3.3. Below the normalized frequency 0.01, the percentage difference between the exact and approximate spectral component values (in dB) of these codes is below 0.23%. Note that the difference between the minimal and maximal asymptotic low-frequency spectral components of these 3/16 3-OZD codes is approximately $10\log_{10}(370/45) \approx 9$ dB.

In Figure 3.18 exact and approximate spectra of 6/32 4-OZD codes with minimal and maximal LFSWs are compared. Using Theorem 3.5, the minimal and maximal LFSWs of 6/32 4-OZD codes are evaluated as $H_{RDS^{(4)}}(0) = 29618.3594$ and $H_{RDS^{(4)}}(0) = 68764.5781$, respectively. This figure demonstrates that the LFSW metric is also a reliable metric for higher order spectral-null codes.
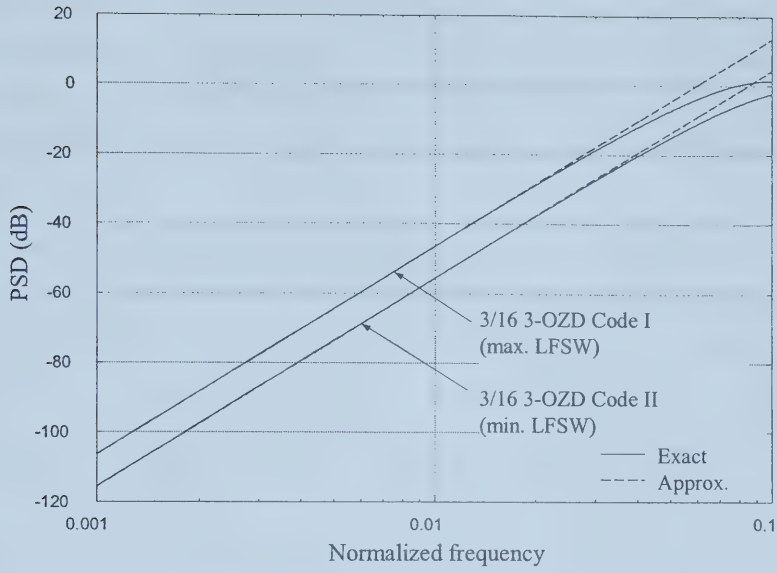
Figure 3.17 Comparison of exact and approximate power spectra of
3/16 dc^3-free codes with minimal and maximal LFSWs.
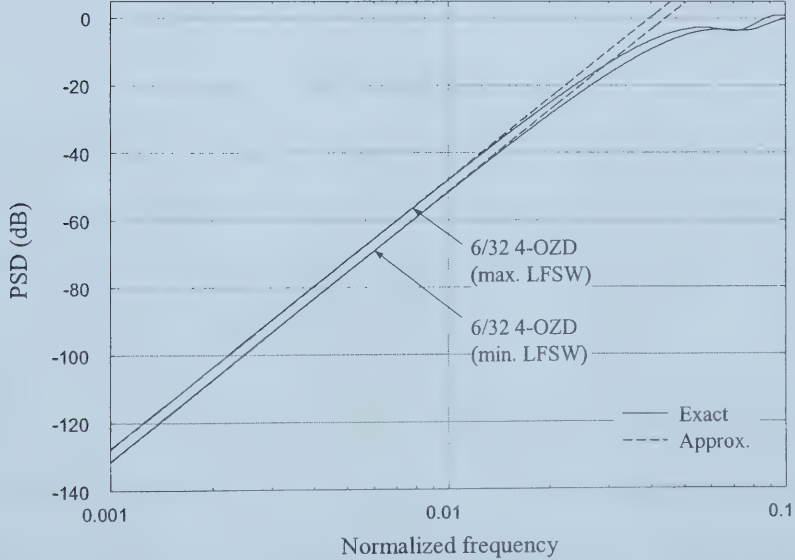
Figure 3.18 Comparison of exact and approximate power spectra of
6/32 dc^4-free codes with minimal and maximal LFSWs.

Theorem 3.5 introduces a convenient method for calculating the LFSW of K-OZD codes in
which neither the closed-form expression of spectrum of a specified code nor knowledge of its

higher derivatives at dc is needed. However, when the set of codewords is large, the effort of listing the required codewords is prohibitive. In the following, a method to enumerate all Kth-order moments of dc^K-free words with length N is presented. Based on this method, the average, minimal, and maximal asymptotic low-frequency spectral components of K-OZD codes can be conveniently obtained.

Enumeration has been employed to calculate the number of codewords of high-order zero-disparity codes [23], and new initial conditions for efficiently enumerating those codewords were considered in Section 2.3.3. Let the codeword $\underline{x}_a = [x_{a,1}, x_{a,2}, \cdots, x_{a,N}]$ be as defined above, let $\boldsymbol{\alpha} = \{\alpha_1, \cdots, \alpha_{N/2}\}$ and $\boldsymbol{\beta} = \{\beta_1, \cdots, \beta_{N/2}\}$ be as defined in Section 2.3.1. Recall that there are two disjoint sets of indices of symbol values in codewords. Also let $J_r = (1/2)\sum_{j=1}^{N} j^r$, $r = 0, 1, \cdots, K$. Therefore, expression (2.31) is satisfied and

$$\sum_{p=1}^{N/2} \alpha_p^r - \sum_{q=1}^{N/2} \beta_q^r = u_a^{(r)}, \ r = 0, 1, \cdots, K.$$

(3.77)

For the dc^K-free word $\underline{x}_a$, combining (2.31) and (3.77) yields:

$$\sum_{p=1}^{N/2} \alpha_p^r = \sum_{q=1}^{N/2} \beta_q^r = (1/2)\sum_{j=1}^{N} j^r = J_r, r = 0, 1, \cdots, K-1$$

$$u_a^{(K)}/2 = \sum_{p=1}^{N/2} \alpha_{a,p}^K - J_K.$$

(3.78)

In enumeration of the number of dc^K-free words, a K-variable generating function is used. Here a $(K+1)$-variable generating function for enumeration of Kth-order moment values of these words is used. Define the enumerating generating function to be:

$$\mathcal{F}_N(v_0, v_1, \cdots, v_K) = \prod_{j=1}^{N}(1 + v_0^{j^0} v_1^{j^1} \cdots v_K^{j^K})$$

$$= \sum_{\lambda_0, \lambda_1, \cdots, \lambda_K} \Lambda_N(\lambda_0, \lambda_1, \cdots, \lambda_K) \cdot v_0^{\lambda_0} v_1^{\lambda_1} \cdots v_K^{\lambda_K}.$$

(3.79)

It can be shown that when $\lambda_r = J_r, r = 0, 1, \cdots, K-1$, a valid codeword $\underline{x}_a$ is enumerated and $\lambda_K = J_K + u_a^{(K)}/2$, from which the Kth-order moment of this codeword can be obtained. From (3.79), it is straightforward to derive the recursive relation:

$$\Lambda_j(\lambda_0, \lambda_1, \cdots, \lambda_K) = \Lambda_{j-1}(\lambda_0, \lambda_1, \cdots, \lambda_K) + \Lambda_{j-1}(\lambda_0 - j^0, \lambda_1 - j^1, \cdots, \lambda_K - j^K),$$

(3.80)

with the initial condition $\Lambda_0(0, \cdots, 0, u_a^{(K)}/2)$.

Figures 3.19-3.22 illustrate the approximate power spectra of K-OZD codes ($1 \leq K \leq 4$ and $N \leq 32$) generated by full-set coding. For comparison, exact spectra of the corresponding K-OZD codes shown in Section 3.1.2 (c) are also included in these figures.
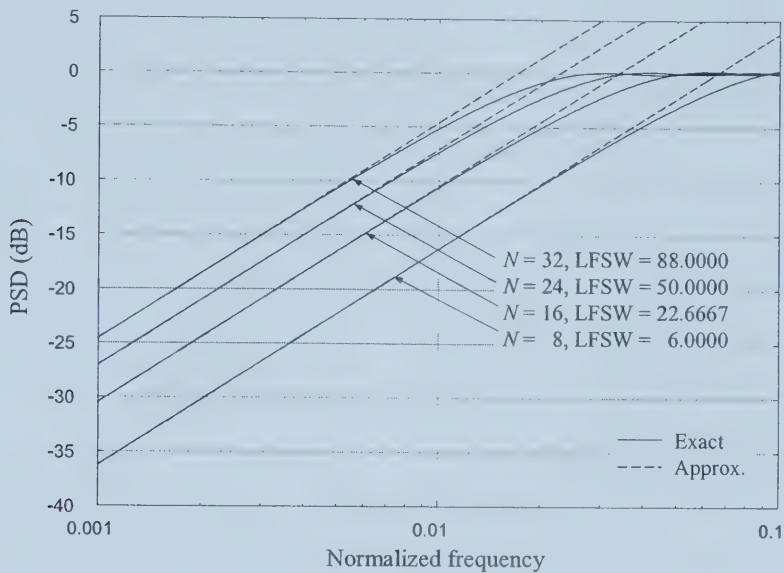
Figure 3.19 Comparison of exact and approximate power spectra of
1-OZD codes generated by full-set coding.
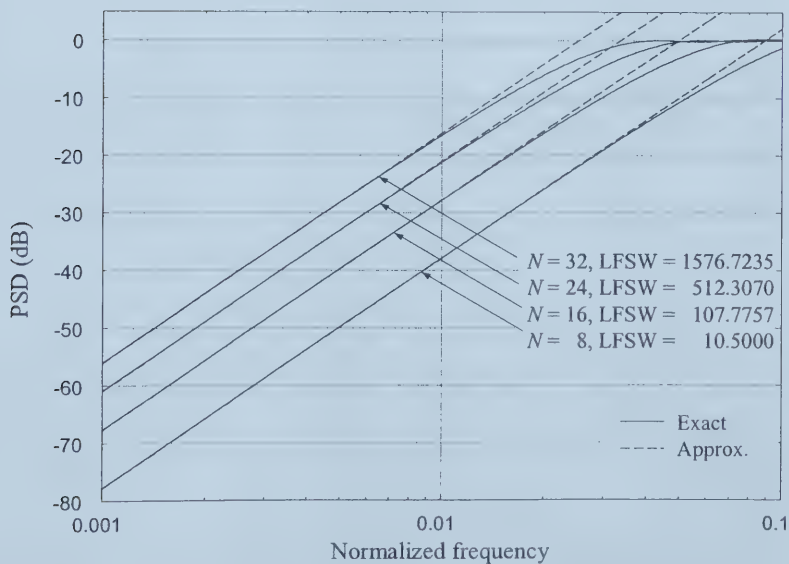
Figure 3.20 Comparison of exact and approximate power spectra of
2-OZD codes generated by full-set coding.
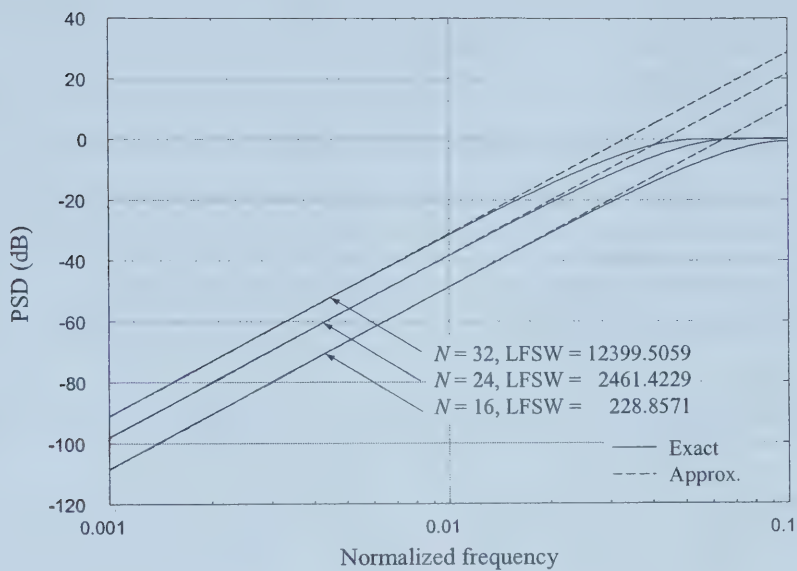
Figure 3.21 Comparison of exact and approximate power spectra of
3-OZD codes generated by full-set coding.
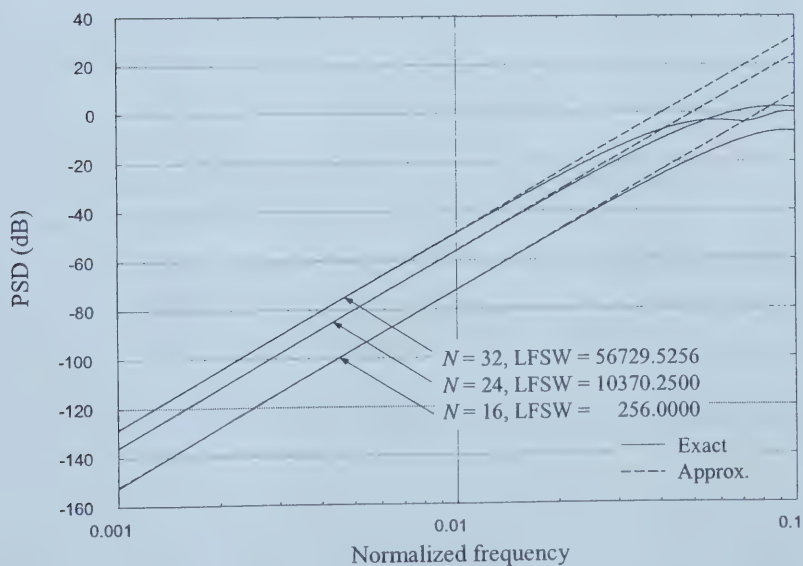
Figure 3.22 Comparison of exact and approximate power spectra of
4-OZD codes generated by full-set coding.

In Table 3.4, minimal and maximal LFSWs of K-OZD codes ($1 \le K \le 4$ and $N \le 32$) are listed. The difference between maximal and minimal LFSW values ranges from zero to over 10 dB. These minimal and maximal values of LFSW of a K-OZD code determine the minimal and maximal asymptotic low-frequency spectral components of this code. However, when the codeword length is relatively large (e.g. $N > 32$) and K is relatively small (e.g. $K = 1$), the size of the codeword set is very large [29] and it is a challenge to sort the moments of all possible codewords. Evaluation of the average LFSW within the full set of codewords is a convenient approach to obtain the average asymptotic low-frequency spectral components which are bounded by the minimal and maximal asymptotic low-frequency spectral components. In Section 3.3.4, simple expressions for LFSW of some specific dc[1]-free codes are presented.

Table 3.4 Minimal and maximal LFSWs of K-OZD codes versus K and codeword length N
($1 \le K \le 4$ and $N \le 32$)

N	1-OZD codes		2-OZD codes		3-OZD codes		4-OZD codes	
	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.
2	0.5000	0.5000						
4	0.5000	2.5000	1.0000	1.0000				
6	1.6667	4.3333						
8	4.2344	6.5625	10.5000	10.5000	8.0000	8.0000		
10	1.5375	16.6625						
12	2.5417	22.1719	8.1042	64.4583	90.7500	90.7500		
14	3.9621	28.1719						
16	5.8939	34.6172	92.5859	110.7227	45.0000	370.0000	256.0000	256.0000
18	8.4168	41.4501						
20	11.6436	48.6408	97.9703	333.1617	300.7250	1448.4500		
22	15.7005	56.1997						
24	20.7125	64.0613	89.1031	899.2086	1394.3691	2842.9082	10370.2500	10370.2500
26	26.8677	72.2418						
28	34.3282	80.7073	326.0984	1274.5938	1862.2751	7619.4888		
30	43.3946	89.4479						
32	54.3465	98.4594	1173.7328	1674.2694	9154.8553	13200.2331	29618.3594	68764.5781

3.3.4 Simple Expressions for LFSW of Specific First-Order Spectral-Null Codes

In this subsection, expressions for LFSWs of 1-OZD codes, low-disparity dc[1]-free codes, and maxentropic dc[1]-free codes are presented. The relationship between the redundancy and the value of LFSW of maxentropic dc[1]-free codes is also discussed.

A. LFSW of 1-OZD codes

The most straightforward method to construct block dc^1-free codes is to employ dc^1-free codewords with an equal number of "+1's" and "−1's" [69]. For ease of calculation, assume that all codewords are used with equal probability. The notation defined in Section 3.3.3 is used, where now $K = 1$.

Theorem 3.6: The LFSW of 1-OZD codes generated by full-set coding is:

$$H_{\mathrm{RDS}^{(1)}}(0) = \frac{1}{12} N(N+1) . \tag{3.81}$$

Proof: Assuming that all codewords are equiprobable, the probability of the codeword $\underline{x}_a$ is $1/A_c$. When $K = 1$, the LFSW of K-OZD codes given in (14) can be expressed as:

$$H_{\mathrm{RDS}^{(1)}}(0) = \frac{1}{N} E\left\{ \left(u_a^{(1)} \right)^2 \right\}, \tag{3.82}$$

where the notation $E\{\ \}$ represents the expectation over all codewords:

$$E\left\{ \left(u_a^{(1)} \right)^2 \right\} = E\left\{ \left(\sum_{j=1}^{N} j x_{a,j} \right)^2 \right\}$$

$$= E\left\{ (x_{a,1})^2 + (2x_{a,2})^2 + \cdots + (N x_{a,N})^2 \right\} + 2 \sum_{j_1=1}^{N-1} \sum_{j_2=j_1+1}^{N} E\left\{ j_1 j_2 x_{a,j_1} x_{a,j_2} \right\},$$

where $1 \le j_1, j_2 \le N$ and $j_1 \ne j_2$. Since $x_{a,j} \in \{-1, +1\}$:

$$E\left\{ \left(u_a^{(1)} \right)^2 \right\} = \frac{1}{6} N(N+1)(2N+1) + 2 \sum_{j_1=1}^{N-1} \sum_{j_2=j_1+1}^{N} E\left\{ j_1 j_2 x_{a,j_1} x_{a,j_2} \right\}$$

$$= \frac{1}{6} N(N+1)(2N+1) + 2 \sum_{j_1=1}^{N-1} \sum_{j_2=j_1+1}^{N} j_1 j_2 E\left\{ x_{a,j_1} x_{a,j_2} \right\}.$$

As shown in [69], the correlation of symbols at different positions within the same codeword is:

$$r_0 = E\left\{ x_{a,j_1} x_{a,j_2} \right\} = -\frac{1}{N-1},$$

and it follows that:

$$2 \sum_{j_1=1}^{N-1} \sum_{j_2=j_1+1}^{N} j_1 j_2 E\left\{ x_{a,j_1} x_{a,j_2} \right\} = 2 r_0 \sum_{j_1=1}^{N-1} \sum_{j_2=j_1+1}^{N} j_1 j_2$$

$$= 2 r_0 \sum_{j_1=1}^{N-1} j_1 \left(\sum_{j_2=1}^{N} j_2 - \sum_{j_2=1}^{j_1} j_2 \right)$$

$$= -\frac{1}{12} N(3N^2 + 5N + 2).$$

By combining this result with (3.82), the theorem follows. $\qquad\square$

Results demonstrating the relationship in (3.81) were previously given in Figure 3.19.

Corollary 3.2: When codeword lengths are large, the absolute difference (in dB) of the average asymptotic low-frequency spectral components of two 1-OZD codes approximately equals twice the absolute difference of the two codeword lengths (in dB).

B.   *LFSW of low-disparity dc¹-free codes*

In addition to 1-OZD codes using zero-disparity words, dc^1-free codes can also be implemented by employing non-zero disparity words. In [1, p.214], the closed-form expression of power spectra of low-disparity codes is given. The codewords with even word length N used in these low-disparity codes are assumed to have disparity of 0 or ± 2. The power spectrum of such low-disparity dc^1-free codes is:

$$H(\omega) = (1-a)\left[1 - \frac{a^2 N^2 F^2(\omega)}{1+a^2+2a\cos(N\omega)}\right]$$ (3.83)

where $a = -1/(N+1)$ and

$$F(\omega) = \frac{\sin(N\omega/2)}{N\sin(\omega/2)}.$$

From (3.83), the LFSW of these codes can be conveniently obtained:

$$H_{\text{RDS}^{(1)}}(0) = \frac{1}{12}(N+2)(N+11).$$ (3.84)

Figure 3.23 demonstrates that approximate PSDs at low frequencies based on the simple expression (3.84) are very close to the exact PSDs obtained by (3.83) at these frequencies.
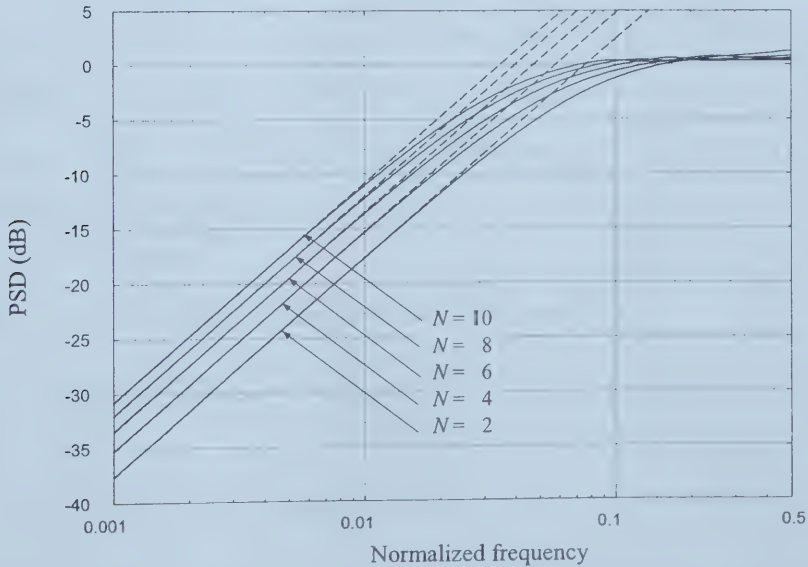
Figure 3.23 Comparison of exact and approximate power spectra of low-disparity dc^1-free codes.

C. LFSW of maxentropic dc¹-free codes

A finite number of $\text{RDS}^{(1)}$ values in a sequence guarantees that there is a first-order spectral null at dc [43]. A maxentropic dc¹-constrained sequence is a sequence with bounded $\text{RDS}^{(1)}$ values that has the maximum entropy probability distribution [33]. For an encoded bipolar sequence $\{x\}$, let $\text{RDS}_l^{(1)}$ be $\text{RDS}^{(1)}$ values at instant l and let $V^- \le \text{RDS}_l^{(1)} \le V^+$. The digital sum variation (DSV), V, is defined as $V = V^+ - V^- + 1$. The capacity of dc¹-free codes is uniquely determined by the value of V [33]. For ease of calculation, the mean of the sequence of $\text{RDS}^{(1)}$ values, $E\{\text{RDS}_l^{(1)}\}$, is generally assumed to be zero [1, p. 192], [62], [80]. Denote B to be the absolute bound of $\text{RDS}^{(1)}$ values where $B = |V^+| = |V^-|$. Given the value of B, the capacity of dc¹-free codes is [33], [80]:

$$C(B) = 1 + \log_2 \cos\frac{\pi}{2(B+1)}, \quad B \ge 1 \tag{3.85}$$

and the corresponding redundancy equals $1 - C(B)$. Recall that in (2.9) the capacity versus sum variation V is given. Similar to (3.50), when the value of B is large, the capacity of maxentropic dc¹-free codes can be approximated as:

$$C(B) \approx 1 - \frac{\pi^2}{8\ln 2}\frac{1}{(B+1)^2}, \quad B \gg 1 \tag{3.86}$$

which is uniquely determined by the value of B.

A closed-form expression for $H_{\text{RDS}^{(1)}}(\omega)$, the power spectrum of the running digital sum sequence $\{\mathbf{RDS}^{(1)}\}$ of maxentropic dc¹-free codes, was presented in Theorem 2 in [80]. Since the LFSW of maxentropic dc¹-free codes is the zero-frequency spectral component of $\{\mathbf{RDS}^{(1)}\}$, it is straightforward to simplify the formula in [80] by setting the frequency ω to be zero. Therefore, the lemma follows.

Lemma 3.1: The LFSW value of maxentropic dc¹-free codes is:

$$H_{\text{RDS}^{(1)}}(0) = \frac{4}{(B+1)^2}\sum_{j=1}^{B}\mu_j\sum_{i=1}^{B}\mu_i\sum_{k=1}^{B}\frac{2\sin(i2k\gamma)\sin(j2k\gamma)}{1 - \frac{\cos(2k\gamma)}{\cos(\gamma)}} - \frac{2}{B+1}\sum_{j=1}^{B}(B+1-j)\sum_{i=1}^{B}\mu_i\sin(i\gamma) \tag{3.87}$$

where $\gamma = \frac{\pi}{2(B+1)}$ and $\mu_b = (B+1-b)\sin(b\gamma)$, $b = 1, \cdots, B$.

In Figure 3.24, exact power spectra of maxentropic dc¹-free codes based on [80] and approximate power spectral of maxentropic dc¹-free codes based on (3.87) are compared.
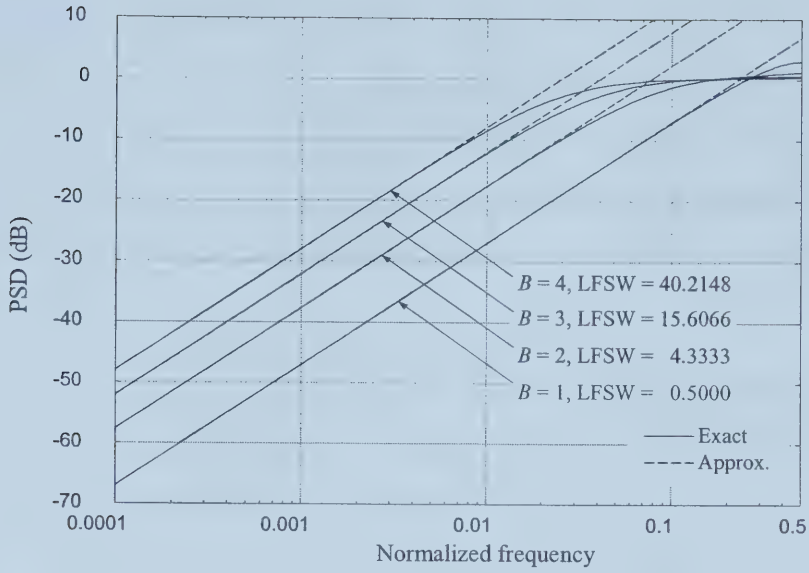
Figure 3.24 Comparison of exact and approximate power spectra of maxentropic dc[1]-free codes.

Figure 3.25 plots the LFSW value versus the redundancy on a logarithmic scale for various values of B. It can be observed that as B increases, redundancy decreases and the LFSW value increases. This figure indicates that for dc[1]-free codes, a deeper spectral null can be obtained at the expense of the code rate. In addition, this figure demonstrates that the logarithms of the LFSW value and redundancy have an approximately linear relationship when B is large motivating the investigation of the asymptotic value of LFSW.
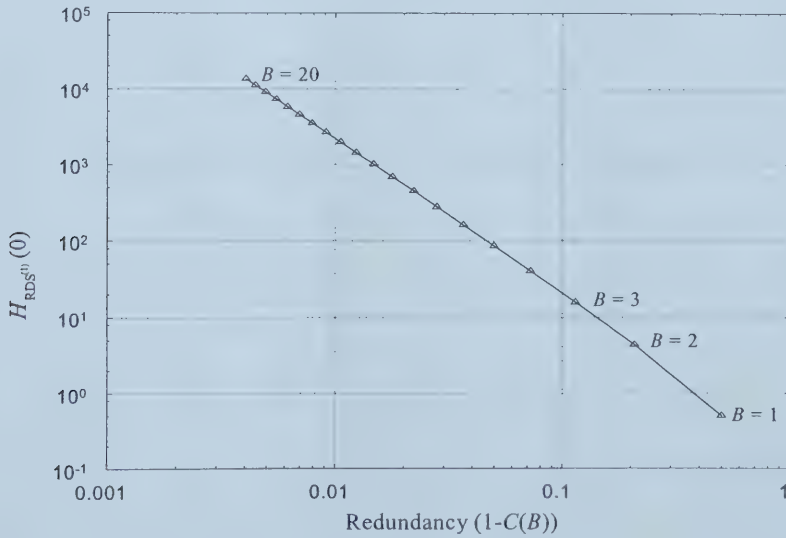
Figure 3.25 LFSW versus redundancy for maxentropic dc[1]-free codes.

Theorem 3.7: When $B \gg 1$, the LFSW value of maxentropic dc^1-free codes is well approximated as:

$$H_{\text{RDS}^{(1)}}(0) \approx \frac{60 - 4\pi^2}{3\pi^4}(B+1)^4 . \tag{3.88}$$

Derivation of this result is deferred to Appendix 2.

Define the product of redundancy and square root of LFSW of maxentropic dc^1-free codes to be the standard product. From (3.86) and (3.88), it can be found that the asymptotic standard product is constant:

$$\lim_{B \to \infty}(1 - C(B))\sqrt{H_{\text{RDS}^{(1)}}(0)} = \frac{\sqrt{3(15 - \pi^2)}}{12 \ln 2} = 0.471660 . \tag{3.89}$$

Based on (3.85) and (3.87), in Table 3.5 the values of capacity and standard product versus B are listed. It can be observed that the percentage difference between the standard product in Table 3.5 and the asymptotic standard product is within 1% when $B \geq 7$.

Table 3.5 Capacity and standard product of maxentropic dc^1-free codes versus the value of B, the absolute bound of RDS$^{(1)}$

B	$C(B)$	$(1-C(B))\sqrt{H_{\text{RDS}^{(1)}}(0)}$
1	0.500000	0.353553
2	0.792481	0.431985
3	0.885777	0.451242
4	0.927603	0.459107
5	0.949984	0.463129
6	0.963367	0.465473
7	0.972009	0.466962
8	0.977914	0.467970
9	0.982128	0.468683
10	0.985240	0.469207
11	0.987604	0.469604
12	0.989443	0.469911
13	0.990900	0.470154
14	0.992075	0.470350
15	0.993036	0.470510
16	0.993833	0.470642
17	0.994500	0.470752
18	0.995064	0.470846
19	0.995546	0.470926
20	0.995960	0.470994
32	0.998365	0.471391
64	0.999579	0.471591
128	0.999893	0.471643
256	0.999973	0.471656
512	0.999993	0.471659
1024	0.999998	0.471660

Chapter 4
MULTIMODE CODES [9]

This chapter begins with consideration of different classes of constrained block codes based on a mapping between a source word and a set of codewords. The guided scrambling (GS) [79] line coding technique is reviewed in Section 4.2. Scrambling polynomials in GS were originally developed in [89]. In this section, scrambling polynomials are further developed, especially for guided scrambling coding with multiple augmenting bits. The result of this further development is used in the implementation of high-order spectral-null multimode codes introduced in Chapter 5. Selection criteria for first-order spectral-null multimode codes are also reviewed.

4.1 Classification of Dc-Free Codes

This chapter begins with consideration of different classes of constrained block codes. Let the source sequence $\{y_n\}$, $y_n \in \{0, 1\}$, be a binary sequence and the coded sequence $\{x_n\}$, $x_n \in \{-1, 1\}$ be a bipolar sequence. Also let M and N be source word length and codeword length respectively. In [21], dc^1-free block codes are categorized as: monomode, bimode, and multimode codes. This classification can also be applied to dc^K-free codes $(K \geq 1)$.

A. Monomode codes

In a monomode code, a source word is related to a codeword through a one-to-one mapping. Monomode coding is the most straightforward method to generate sequences with a spectral null at zero frequency. Both the encoder and decoder in monomode coding are state-independent. Let $\underline{y}_a$ be a source word of length M and $\underline{x}_a$ be the corresponding codeword of length N, $a = 1, 2, \cdots, 2^M$. Figure 4.1 illustrates the mapping between source words and codewords in monomode codes.
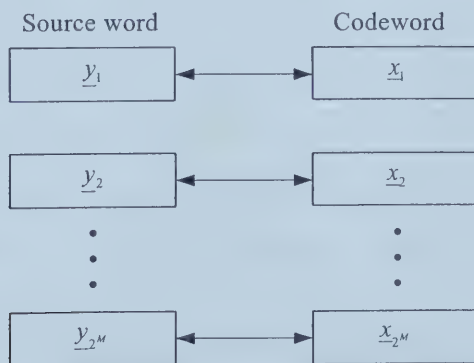
Figure 4.1 Mapping between source words and codewords in monomode codes.

Kth-order zero-disparity codes ($K \geq 1$) defined in Section 2.3 are dc^K-free monomode codes. Each codeword in these codes is a dc^K-free word. From the definition (2.15), in each dc^K-free word the number of "−1's" equals the number of "+1's". Methods of efficiently encoding and decoding dc^1-free monomode codes remain an area of active research [59, p. 530], [81]. Table look-up is an efficient way to generate K-OZD codes for relatively small values of source word length M, for example, $1 \leq M \leq 12$ for 2-OZD codes [30]. However, the complexity of encoding and decoding of look-up tables increases exponentially with an increase in codeword length [82] resulting in this direct coding method to often be impractical [21] especially for dc^K-free monomode codes with high rate and a low order spectral-null at dc. An efficient coding method, enumerative coding, has been proposed for dc^1- and dc^2-free codes in [82] and [22] respectively, and has been demonstrated to be a promising scheme for generation of dc^1-free runlength-limited codes with high rate and good power spectrum performance [20]. However, enumerative coding is considered less practical for efficient high-order spectral-null codes since a large amount of information needs to be precomputed and stored [27]. Other coding schemes have been designed in the literature for high-rate dc^K-free monomode codes ($K \leq 3$) with little consideration of the rejection of low-frequency components given the order of spectral null. These include schemes in [81], [83], [84] for dc^1-free codes, in [27], [30] for dc^2-free codes, and in [28] for dc^3-free codes. Coding methods presented in [24], [26], [27] for higher orders K of spectral null have been shown to be inefficient [28]. A general method to obtain good PSD performance of a dc^K-free monomode code ($K \geq 1$) has been discussed in Section 3.3.3.

B. Bimode codes

Compared to monomode coding, for the same codeword length bimode coding [16] can realize higher-rate dc^1-free codes. In bimode codes, each source word is mapped into one or two corresponding codewords (note that at least one source word has two channel representations) where $\underline{x}_a$ represents the unique codeword mapped into the source word $\underline{y}_a$ and $\{\underline{x}_a(1), \underline{x}_a(2)\}$ represents two codeword candidates (the codeword pair) $\underline{x}_a(1)$ and $\underline{x}_a(2)$ mapped into the source word $\underline{y}_a$, $a = 1, 2, \cdots, 2^M$. According to a given coding rule, one of these two candidates is selected as the codeword to be sent to the constrained channel. The mapping relationship of bimode codes is shown in Figure 4.2.

In an alternating code [16], a dc^1-free bimode code, if a source word is mapped to a unique codeword, this codeword must be zero-disparity word; if mapped to codeword pair, both codeword candidates cannot be positive-disparity or both cannot be negative-disparity words.

Codewords are selected from the codeword pairs in order to minimize the absolute $RDS^{(1)}$ value at the end of the transmitted word [16]. In this thesis, this criterion is called the minimum $RDS^{(1)}$ ($MRDS^{(1)}$) selection criterion.[10] By using this alternating code, the accumulated running digital sum values of the coded sequence can be bounded resulting in a dc^1-free sequence [43].

Source word   Codeword

$y_1 \longleftrightarrow \underline{x}_1$ or $\{\underline{x}_1(1), \underline{x}_1(2)\}$

$y_2 \longleftrightarrow \underline{x}_2$ or $\{\underline{x}_2(1), \underline{x}_2(2)\}$

$\vdots \qquad\qquad\qquad\qquad \vdots$

$y_{2^M} \longleftrightarrow \underline{x}_{2^M}$ or $\{\underline{x}_{2^M}(1), \underline{x}_{2^M}(2)\}$
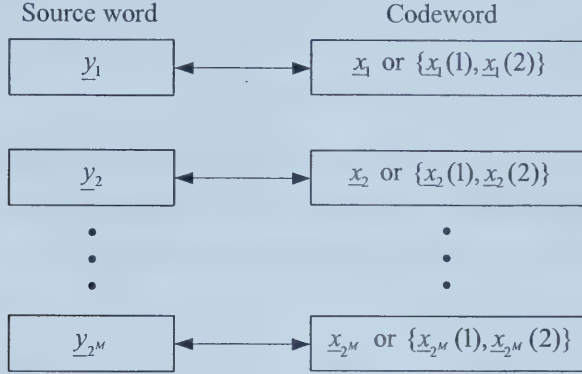
Figure 4.2 Mapping between source words and codewords in bimode codes.

Similar to the alternating code, two example dc^2-free bimode codes are constructed in [22], [26], and were shown previously in Examples 3.3 and 3.4 of this thesis. When a source word is mapped to only one codeword, this codeword needs to be a dc^2-free word; when mapped to a codeword pair, the codeword candidates must have zero value of $RDS^{(1)}$ and different signs of $RDS^{(2)}$ values. During encoding, codewords from the codeword pairs are selected to minimize the absolute $RDS^{(2)}$ value at the end of each transmitted word. Bounded $RDS^{(2)}$ values of the coded sequence guarantee this sequence to be a dc^2-free sequence [25], [26]. Since the codeword candidates in all codeword pairs are not necessarily dc^2-free words, the coded words need to be correlated in order to make the whole coded sequence meet the channel requirement. Bimode encoding is state-dependent whereas bimode decoding is state-independent. A general method to construct dc^K-free bimode codes ($K > 1$) will be introduced in Chapter 5.

Although the scheme described above is an efficient bimode coding method, the encoding and decoding mapping can generally only be implemented through look-up tables. As exceptions to this method, two kinds of dc^1-free bimode codes, the polarity switch code [85], [86] and guided scrambling dc^1-free codes (with one redundant bit) [79], are constructed without look-up tables. For convenience, in these two systematic coding schemes, each source word is mapped into a codeword pair. Based on the $RDS^{(1)}$ value at the end of the last codeword, a polarity switch encoder balances the coded sequence by either directly sending the input source

---

[10] Note that different notations regarding this selection criterion can be found in [21] and [79].

word ($N-1$ source bits) with an appended polarity bit to the channel, or switching the source word to its complementary word (in which logic ones are replaced with logic zeros, and vice versa) to which the opposite polarity bit is appended. Essentially, polarity switch coding maps each source word to a complementary codeword pair. The code rate of a polarity switch code is:

$$R = (N-1)/N . \tag{4.1}$$

The sum variance of this code is [1, p. 217]:

$$\sigma^2_{\text{RDS}^{(1)}} = (2N-1)/3 . \tag{4.2}$$

Bimode coding can result in efficient dc[1]-free codes. However, when both rate efficiency and spectrum performance (based on cutoff frequency or equivalently, sum variance) are considered, it has been shown that the polarity switch code does not approach the performance of the ideal maxentropic code [21].

C. Multimode codes

Multimode coding is an extension of bimode coding. As shown in Figure 4.3, in multimode codes, each source word y_a ($a = 1, 2, \cdots, 2^M$) is represented by a set of codewords $C_a = \{x_a(1), \cdots, x_a(|C_a|)\}$ where $|C_a|$ denotes the number of codewords in C_a ($|C_a| \geq 1$) and the maximum $|C_a|$ value is larger than two.

Source word   Codeword

y_1 ⟷ $\{x_1(1), \cdots, x_1(C_1)\}$

y_2 ⟷ $\{x_2(1), \cdots, x_2(C_2)\}$

y_{2^M} ⟷ $\{x_{2^M}(1), \cdots, x_{2^M}(C_{2^M})\}$
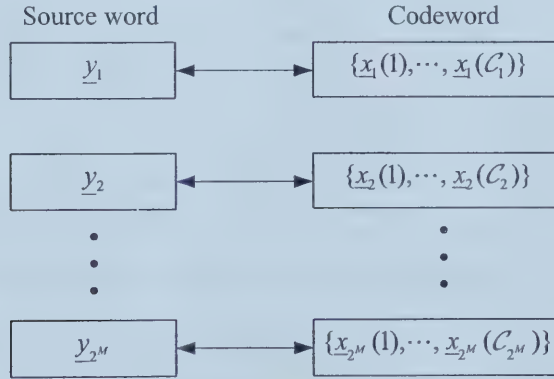
Figure 4.3 Mapping between source words and codewords in multimode codes.

There are three main coding approaches that implement mappings between source words and the corresponding codeword sets in multimode codes in the literature: guided scrambling [79], dc[1]-free coset coding [87], and the scrambling of a Reed-Solomon (RS) code [88].

In dc[1]-free multimode codes, it is required that there exists at least a pair of codewords with a different sign of disparity in each selection set. Furthermore, for better spectrum performance it is desirable that each selection set have as many such codeword pairs as possible. Therefore, for

a given codeword length, multimode codes offer more possibilities than bimode codes for selection of a "good" codeword from a selection set. The drawback is a loss of rate efficiency. With appropriate selection criteria, when both rate and spectrum performance are considered, dc^1-free multimode codes have been shown to be advantageous over dc^1-free bimode codes [21]. Alternative selection criteria in dc^1-free multimode codes will be discussed in Section 4.2.3. Selection criteria for dc^K-free multimode codes ($K > 1$) will be proposed and evaluated in Chapter 5.

Since guided scrambling is easy to implement [79] and well-developed [89]-[91], guided scrambling has been employed as a primary coding technique in application of dc^1-free multimode codes [21]. In the next section, the structure, encoding and decoding mechanisms, and scrambling polynomials for guided scrambling will be discussed. Based on this knowledge of guided scrambling, in Chapter 5 guided scrambling dc^K-free multimode codes ($K > 1$) that demonstrate superior performance over other dc^K-free codes in the literature are constructed.

4.2   Guided Scrambling (GS)

Guided scrambling [79] is an extension of the well-known self-synchronizing scrambling technique. Its encoder involves augmenting the source words, scrambling the augmented words to form a set of quotients, and selecting the appropriate encoded word from the quotient selection set. Its decoder is very simple: decoding is performed by unscrambling the received word and discarding the augmenting bits.

The scrambling and unscrambling processes in GS are trivial. The arithmetic operations in GS are from ring of polynomials over the Galois field of order two, referred to GF(2) [92]. The complexity of GS coding resides in quotient selection. Both the scrambling polynomial and selection criteria affect the power spectrum of the coded sequence.

4.2.1   Structure of GS Encoder and Decoder

It is convenient to employ polynomial representations for the description of the GS coding process. Let a binary word $\underline{b} = (b_{J-1}, b_{J-2}, \cdots, b_1, b_0)$ of length J ($J \geq 1$) be represented by the polynomial:

$$b(x) = b_{J-1}x^{J-1} + b_{J-2}x^{J-2} + \cdots + b_1 x + b_0 \tag{4.3}$$

where $b_j \in \{0,1\}$ is the value of the bit in position j and $J - 1$ represents the most significant position in the word. The encoding and decoding mechanisms of GS are presented in [79]. In each encoding interval, the source word $s(x)$ of length I is preceded by all A-bit binary patterns

$a_i(x)$, $i = 0, 1, \cdots, 2^A - 1$, to generate the augmented words $v_i(x) = a_i(x)x^I + s(x)$ of length $N = I + A$. A quotient selection set $\mathcal{Q}$ is obtained by scrambling these augmented words. The quotient $q_i(x)$ corresponding to the augmented word $v_i(x)$ is:

$$q_i(x) = Q_{d(x)}\left[x^D v_i(x)\right] \tag{4.4}$$

where the operator $Q_{d(x)}[\cdot]$ denotes the formation of a quotient through modulo-2 division of its argument by the scrambling polynomial $d(x)$ of degree D:

$$d(x) = x^D + d_{D-1}x^{D-1} + \cdots + d_1 x + 1. \tag{4.5}$$

Note that for ease of discussion and without loss of generality, the most and least significant coefficients of scrambling polynomials are defined to be 1 [89]. Based on the given selection criterion, a quotient $q(x) \in \mathcal{Q}$ is selected as the codeword to be sent to the channel. Note that if $A = 1$, a bimode code is constructed; if $A > 1$, a multimode is constructed.

In the GS decoder, the decoded augmented word $\hat{v}(x)$ is obtained by multiplying the received word $\hat{q}(x)$, which is the output of the channel, by the scrambling polynomial $d(x)$ used in the encoder and discarding the D least significant bits of $\hat{q}(x)d(x)$:

$$\hat{v}(x) = Q_{x^D}\left[\hat{q}(x)d(x)\right]. \tag{4.6}$$

If there are no transmission errors at the output of the channel, $\hat{q}(x) = q(x)$, $\hat{v}(x) = v(x)$, and the GS decoder recovers the source word correctly by recovering the augmenting bits from $\hat{v}(x)$. Otherwise, let $\hat{q}(x) = q(x) + e(x)$ where $e(x)$ is an error pattern at the output of the channel. In this error pattern, a coefficient of "1" for the term x^i means that there is an error at position i. From (4.6), it can be obtained that:

$$\hat{v}(x) = Q_{x^D}\left[(q(x) + e(x))d(x)\right]$$

$$= v(x) + Q_{x^D}\left[e(x)d(x)\right]. \tag{4.7}$$

Removing the A augmenting bits from $\hat{v}$ results in the decoded word $\hat{s}(x)$. Expression (4.7) shows that at the output of the GS decoder, errors at the input of the decoder are extended during decoding, and this extension is upper bounded by the weight of $d(x)$. When GS is used in a noisy channel, to prevent large error extension, it is desirable to choose a scrambling polynomial with small weight [89].

Depending on the mechanism of augmenting the source word with A redundant bits, guided scrambling encoding can be categorized as *parallel GS encoding* and *serial GS encoding*. In each encoding interval, the parallel GS encoders augment all 2^A binary patterns to a source

word at the same time, whereas the serial GS encoders augment all 2^A binary patterns in serial. Figure 4.4 depicts parallel and serial GS encoding. The coded sequences generated by both forms of encoders can use the same decoder for decoding.

Guided scrambling can also be categorized as block coding and continuous coding due to the different shift register updating procedure in GS coding [79], [89]. In block GS encoding and decoding, all shift registers are cleared after the consideration of each word. Implementation of block GS coding is easier than continuous GS coding, and it has been shown that block GS decoding has less error extension than continuous GS decoding [79]. In addition, when source symbols are independent and equiprobable, the power spectrum of coded sequences generated through continuous GS and block GS with the same scrambling polynomial and selection criterion are identical [89], [90]. Since only the case of independent and equiprobable source symbols is considered in this thesis, in the remainder of this thesis, block GS is employed. Block GS encoding and decoding are illustrated in Figure 4.4.
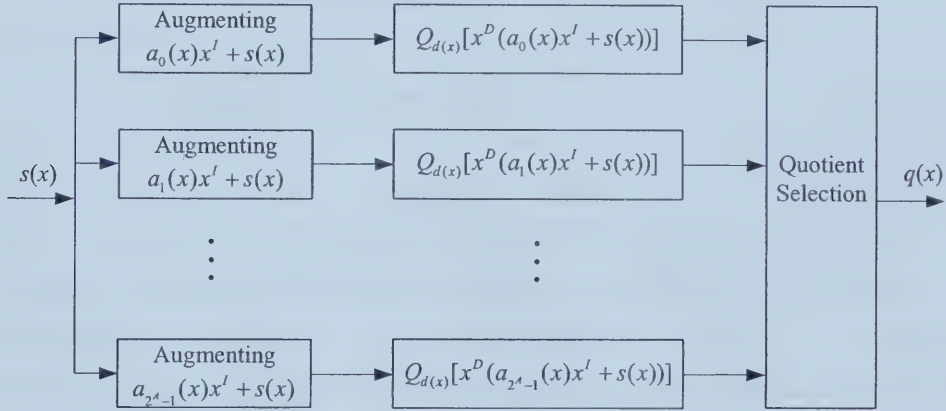
Figure 4.4 (a) Parallel block guided scrambling encoding.
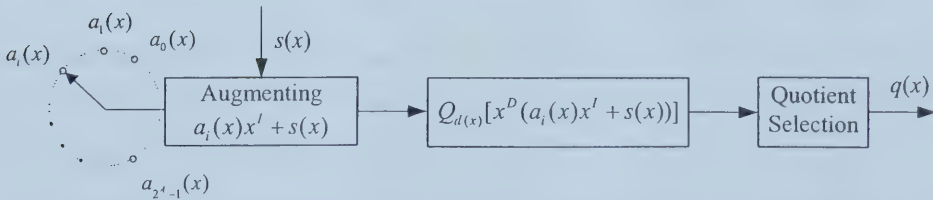
Figure 4.4 (b) Serial block guided scrambling encoding.
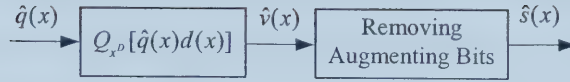
Figure 4.4 (c) Block guided scrambling decoding.

4.2.2   Scrambling Polynomials

In guided scrambling, scrambling polynomials play a key role in determining the mapping between source words and quotient (codeword) selection sets, and have been developed in [79], [89], where the emphasis has been on systems using a single augmenting bit per word. In GS, when one redundant bit is augmented to all source words and the scrambling polynomial is $d(x) = x + 1$, it is straightforward to show that each source word is mapped to a complementary codeword pair [79], [89]. Assuming that source words of the encoder considered are independent and equiprobable and assuming that the MRDS$^{(1)}$ selection criterion is employed, the power spectrum of this GS code is identical to that of a polarity switch code given the same codeword length.

GS codes using more than one augmenting bit per word belong to the class of multimode codes. Due to the property that the disparity of complementary words cannot both be positive or both be negative, it is sufficient for construction of dc^1-free codes if every quotient selection set in GS is composed of complementary quotients [21], [79], [89]. Therefore, it is crucial to know which scrambling polynomials will generate complementary quotients in every selection set. For the case of A redundant bits in GS, it has been shown that $d(x) = x^A + 1$ is an appropriate scrambling polynomial [79]. However, improved performance might result from use of other polynomials, therefore other scrambling polynomials for general cases need to be determined. In the following, necessary and sufficient conditions for scrambling polynomials (for arbitrarily large selection sets) to ensure generation of complementary quotients in the selection sets are derived. The relationship between augmenting bit patterns and complementary quotients is also discussed. This result on scrambling polynomials is used in Chapter 5 of this thesis for construction of dc^K-free multimode codes.

Let GS coding have polynomial representations as discussed in the last subsection. Denote base polynomials $d_B(x)$ to be scrambling polynomials of degree D_B. Extension of base polynomials to families of polynomials is discussed in [89] and given by Lemma 4.1 below.

Lemma 4.1 [89]: In guided scrambling, all polynomials with degree less than or equal to $N - 1$ (N is the quotient length) are base polynomials for different families of scrambling

polynomials, and polynomials in the same family generate the same quotient selection sets but have different associated remainders. A scrambling polynomial $d_F(x)$ of degree D_F is in the same family of polynomials as $d_B(x)$ if:

$$d_F(x) = d_B(x)x^L + p(x) \tag{4.8}$$

where $D_B \leq N-1$, $L \geq N - D_B$, and $p(x)$ is a polynomial of degree at most $D_F - N$ with the the least significant coefficient equal to one.

The Theorem 4.1 below establishes the properties of base polynomials that ensure the desired quotient set characteristics. In the proof of this theorem, the following well-known result is required.

Lemma 4.2 [93]: A polynomial $p(x)$ is divisible by $x+1$ if, and only if, it has even weight.

Theorem 4.1: Let $d_B(x)$ be a base scrambling polynomial with degree D_B, and let A be the number of augmenting bits that precede each source word. Complementary quotients appear in all quotient selection sets if, and only if, $D_B \leq A$ and $d_B(x)$ has even weight.

Proof of sufficiency: When $A=1$, $d_B(x) = x+1$ generates complementary quotients [79], [89]. Consider formation of quotients when $A > 1$. Suppose that $d_B(x) = d_1(x)d_2(x)$, where the degrees of $d_1(x)$ and $d_2(x)$ are D_1 and D_2 ($D_1, D_2 \geq 1$) respectively. Then $D_B = D_1 + D_2$ and:

$$q_i(x) = Q_{d_1(x)d_2(x)}\left[x^{D_1}x^{D_2}v_i(x)\right]$$

$$= Q_{d_1(x)}\left\{x^{D_1}Q_{d_2(x)}\left[x^{D_2}v_i(x)\right]\right\}. \tag{4.9}$$

Let the intermediate quotients from two different augmented words $v_i(x)$ and $v_j(x)$ be $Q_{d_2(x)}\left[x^{D_2}v_i(x)\right]$ and $Q_{d_2(x)}\left[x^{D_2}v_j(x)\right]$, and consider the case when $D_B \leq A$. If the $D_2 + 1$ most significant bits of $v_i(x)$ and $v_j(x)$ are related through modulo-2 addition by the pattern of coefficients in $d_2(x)$, and all other bits in $v_i(x)$ and $v_j(x)$ are identical (there are 2^{A-1} such pairs in each set of augmented words) then $Q_{d_2(x)}\left[x^{D_2}v_i(x)\right]$ and $Q_{d_2(x)}\left[x^{D_2}v_j(x)\right]$ differ in only the most significant bit. That is, these two quotients are related through mod-2 addition by x^{N-1}.

As shown in [89]:

$$Q_{x+1}\left[x \cdot x^{N-1}\right] = x^{N-1} + x^{N-2} + \cdots + x + 1. \tag{4.10}$$

Letting $d_1(x) = x+1$ in (4.9) yields:

$$q_i(x) = Q_{x+1}\left\{x \cdot Q_{d_2(x)}\left[x^{D_2} v_i(x)\right]\right\}$$

$$= Q_{x+1}\left\{x \cdot Q_{d_2(x)}\left[x^{D_2} v_j(x)\right]\right\} + Q_{x+1}\left\{x \cdot x^{N-1}\right\}$$

$$= \overline{q_j(x)}.$$

Therefore when $D_B \leq A$ and $d_B(x)$ is divisible by $x+1$, all quotient sets contain 2^{A-1} pairs of complementary quotients. Since $d_B(x)$ is divisible by $x+1$, it has even weight.

Proof of necessity: Let an arbitrary source word of length I be:

$$s(x) = s_{I-1} x^{I-1} + s_{I-2} x^{I-2} + \cdots + s_1 x + s_0 \tag{4.11}$$

and let an arbitrary augmenting word of length A be:

$$a(x) = a_{A-1} x^{A-1} + a_{A-2} x^{A-2} + \cdots + a_1 x + a_0. \tag{4.12}$$

Let the scrambling polynomial be as defined in (4.5) and let $q_i(x)$ and $q_j(x)$ be the quotients corresponding to the source word augmented by bit patterns $a_i(x)$ and $a_j(x)$, respectively:

$$q_i(x) = Q_{d(x)}\left[x^D\left(x^I a_i(x) + s(x)\right)\right] \tag{4.13}$$

$$q_j(x) = Q_{d(x)}\left[x^D\left(x^I a_j(x) + s(x)\right)\right]. \tag{4.14}$$

From linearity of division, it follows that:

$$q_i(x) + q_j(x) = Q_{d(x)}\left[x^D(x^I a_i(x) + s(x))\right] + Q_{d(x)}\left[x^D(x^I a_j(x) + s(x))\right]$$

$$= Q_{d(x)}\left[x^{I+D} a_i(x)\right] + Q_{d(x)}\left[x^{I+D} a_j(x)\right]$$

$$= h_i(x) + h_j(x). \tag{4.15}$$

Let $r_i(x)$ and $r_j(x)$ be the remainders associated with the quotients $h_i(x)$ and $h_j(x)$ respectively. From (4.15), then:

$$\left[h_i(x) + h_j(x)\right] \cdot d(x) = x^{I+D}\left[a_i(x) + a_j(x)\right] + \left[r_i(x) + r_j(x)\right]. \tag{4.16}$$

Extending the right hand side of (4.16), it is observed that coefficients for the consecutive terms x^D up to x^{I+D-1} must be zero because the degrees of the remainders are less than or equal to $D-1$. These zero coefficients force coefficients of similar powers of x on the left hand side of this equation to be zero also.

If quotients $q_i(x)$ and $q_j(x)$ are to be complementary, it is clear from (4.15) that $h_i(x)$ and $h_j(x)$ must be complementary, and the sequence $h_i(x) + h_j(x)$ is the all-one sequence of length

$N = I + A$. Coefficients on the left hand side of (4.16) are constructed through multiplication of this all-one sequence by $d(x)$ ($D \geq 2$):

$$(x^{I+A-1} + x^{I+A-2} + \cdots + x + 1) \cdot d(x)$$

$$= x^{I+A+D-1} +$$

$$x^{I+A+D-2}(1 \oplus d_{D-1}) +$$

$$x^{I+A+D-3}(1 \oplus d_{D-1} \oplus d_{D-2})$$

$$+ \cdots +$$

$$x^{I+D}(1 \oplus d_{D-1} \oplus d_{D-2} \oplus \cdots \oplus d_{D-A+1}) +$$

$$x^{I+D-1}(1 \oplus d_{D-1} \oplus d_{D-2} \oplus \cdots \oplus d_{D-A+1} \oplus d_{D-A})$$

$$+ \cdots +$$

$$x^{D}(1 \oplus d_{D-1} \oplus d_{D-2} \oplus \cdots \oplus d_{D-A+1} \oplus d_{D-A} \oplus \cdots \oplus d_{D-A-I+1}) +$$

$$x^{D-1}(1 \oplus d_{D-1} \oplus d_{D-2} \oplus \cdots \oplus d_{D-A+1} \oplus d_{D-A} \oplus \cdots \oplus d_{D-A-I+1} \oplus d_{D-A-I})$$

$$+ \cdots +$$

$$x^{I+A-1}(d_{I+A-1} \oplus d_{I+A-2} \oplus \cdots \oplus d_1 \oplus 1) +$$

$$x^{I+A-2}(d_{I+A-2} \oplus \cdots \oplus d_1 \oplus 1)$$

$$+ \cdots +$$

$$x(d_1 \oplus 1) + 1$$

where the operation "$\oplus$" represents modulo-2 addition. The requirement for zero-valued coefficients in the terms x^{D} through x^{I+D-1} places restrictions on the coefficients of $d(x)$. It is straightforward to verify that to generate these zero-valued coefficients, it is necessary that the scrambling polynomials have the following form:

i)   when $D = 1$, $d(x) = x + 1$;

ii) when $2 \leq D \leq A$, $d(x) = x^D + d_{D-1}x^{D-1} + \cdots + d_1 x + 1$, where

   $d_{D-1} \oplus d_{D-2} \oplus \cdots \oplus d_1 = 0$;

iii) when $A + 1 \leq D \leq N - 1$, no polynomials satisfy the conditions;

iv) when $D \geq N$, $d(x) = x^D + d_{D-1}x^{D-1} + \cdots + d_1 x + 1$, where

   $d_{D-1} \oplus d_{D-2} \oplus \cdots \oplus d_{D-A} = 1$,

   $d_j = 0$ for $j = D - I - A + 1, \ D - I - A + 2, \ \cdots, \ D - A - 1$, and there are no restrictions

   on d_j for $j = 1, 2, \cdots, D - I - A$.

From *i)* and *ii)*, it is evident that if $D \leq A$, $d(x)$ must have even weight. Further, *iii)* and *iv)* impose the structure for families of polynomials. Therefore, for generation of complementary quotients, the base polynomial must have degree less than or equal to A and must have even weight. $\qquad\square$

Corollary 4.1: Complementary quotients will be generated for augmenting bit patterns in which the most significant $D_2 + 1$ bits are related through modulo-2 addition by $Q_{x+1}[d_B(x)]$ and the least significant $A - D_2 - 1$ bits are identical.

Corollary 4.2: The remainders corresponding to the complementary quotients generated by the base polynomial $d_B(x)$ are related through modulo-2 addition by $Q_{x+1}[d_B(x)]$.

Proof: Let $d_B(x) = (x+1)d_2(x)$ be a base polynomial of degree D_B and let the degree of $d_2(x)$ be D_2. Then $D_B = D_2 + 1$. Let $q_i(x)$ and $q_j(x)$ be a complementary quotient pair in a selection set, let $a_i(x)$ ($r_i(x)$) and $a_j(x)$ ($r_j(x)$) be the corresponding augmenting words (remainders) respectively. From (4.15), there exists:

$$x^{I+D_B}\left[a_i(x) + a_j(x)\right] = \left[q_i(x) + q_j(x)\right]d_B(x) + \left[r_i(x) + r_j(x)\right]. \qquad (4.17)$$

Furthermore, from Theorem 4.1 and Corollary 4.1, it can be found that:

$$x^{I+A}d_2(x) = \left[q_i(x) + q_j(x)\right](x+1)d_2(x) + \left[r_i(x) + r_j(x)\right]$$
$$= (x^N + 1)d_2(x) + \left[r_i(x) + r_j(x)\right].$$

Since $N = I + A$ and $D_2 < N$, the corollary follows. $\qquad\square$

Note that in block GS coding, the remainders are cleared at the end of an encoding interval. As a result, with block codes, coded sequences generated through all polynomials in a family have the same power spectrum, and it is not necessary to employ a polynomial other than the base polynomial in block GS coding.

Corollary 4.3: There are 2^{A-1} base scrambling polynomials that will generate complementary quotients in each quotient selection set.

Lemma 4.3 [58, p. 3]: $C_0^n + C_2^n + C_4^n + \cdots = 2^{n-1}$.

From Theorem 4.1 and Lemma 4.3, Corollary 4.3 can be conveniently proved. For a given augmenting bit number A, the total number of base polynomials equals:

$$1 + 1 + \sum_{D_B=3}^{A}\left(C_0^{D_B-1} + C_2^{D_B-1} + C_4^{D_B-1} + \cdots\right)$$

$$= 1 + 2^0 + \sum_{D_B=3}^{A} 2^{D_B-2} = 2^{A-1}.$$

Example 4.1: The base polynomial $d_B(x) = (x+1)(x+1) = x^2 + 1$ will generate complementary quotients in quotient selection sets if $A \geq 2$. Consider the case when $I = 7$ and $A = 3$. For an arbitrary source word, for instance, the all-zero word, the quotient set is listed in Table 4.1. Clearly, $2^{3-1} = 4$ complementary pairs of quotients are generated in the quotient selection set, and complementary quotients correspond to augmenting bit patterns whose least significant bit is identical and whose first two bits are related through summation by $Q_{x+1}[x^2 + 1] = x + 1$. In addition, reminders corresponding to the complementary quotients are also related through summation by $Q_{x+1}[x^2 + 1] = x + 1$.

Table 4.1 Quotient selection set for the all-zero source word
when $I = 7$, $A = 3$ and $d_B(x) = x^2 + 1$

Augmenting bit patterns	Quotients	Remainders
000	0000000000	00
001	0010101010	10
010	0101010101	01
011	0111111111	11
100	1010101010	10
101	1000000000	00
110	1111111111	11
111	1101010101	01

4.2.3 Characteristics of GS Coding

Characteristics of guided scrambling have been extensively developed in [79], [89]-[91]. Some of these characteristics used in this thesis are reviewed below.

Characteristic 4.1: Assume that a GS encoder is driven by a source sequence composed of stationary, memoryless, and equiprobable binary symbols. In this case, the power spectra of block GS and continuous GS coded sequences are identical given the same scrambling polynomial and selection criterion.

Define the complement of a quotient set Q to be the set $\overline{Q}$ that contains all complements of the quotients in the quotient set Q.

Characteristic 4.2: Every quotient set has a complement. Scrambling polynomials of even weight result in every quotient set being its own complement.

Assume that states in a GS dc^1-free encoding are represented by RDS$^{(1)}$ values at the end of each codeword. Let complementary states be the states in which the differences between these word-end RDS$^{(1)}$ values and the center of the Markov chain in the encoder have the same absolute value but opposite signs.

Characteristic 4.3: Given the MRDS$^{(1)}$ selection criterion and scrambling polynomials of even weight, in complementary states, complementary quotients are selected resulting in the next states being complementary.

In the next chapter, the guided scrambling coding technique is employed in a new class of dc^K-free multimode codes $(K \geq 1)$.

4.2.4   Selection Criteria for Dc^1-Free Multimode Codes

In a multimode code, each source word has multiple channel representations permitting the multimode encoder to select the "best" representation from this selection set as the codeword. Assume that the scrambling polynomial has a form discussed in the last section and realizes an appropriate mapping between source words and codewords. Selection criteria determine how to select a codeword resulting in a coded sequence with good performance from those selection sets. As described in Chapter 1, it is desirable that the power spectrum of a dc-free code has a large rejection of low frequencies.

First-order running digital sum of coded sequences plays an important role in power spectrum of dc^1-free codes. In Chapter 3, it was shown that given digital sum variation, sum variance and LFSW of a maxentropic dc^1-free sequence can be determined, and the larger the DSV value, the less suppression of low-frequency components. From their definitions, sum variance and LFSW of a sequence with a first-order spectral-null at dc are statistics that reflect the RDS$^{(1)}$ in the sequence. Below, two well-known selection criteria for dc^1-free multimode codes which are related to RDS$^{(1)}$ are reviewed. Power spectra of GS dc^1-free multimode codes using these criteria are evaluated.

As defined in Section 3.1.2, the encoded symbol sequence $\{x\} = \{x_1, x_2, \cdots, x_l, \cdots\}$ is assumed to be generated through concatenation of length-N codewords $\{\underline{x}\} = \{\underline{x}_1, \underline{x}_2, \cdots, \underline{x}_n, \cdots\}$ where $\underline{x}_n = [x_{n,1}, x_{n,2}, \cdots, x_{n,N}]$, and the relationship between x_l and $x_{n,j}$ is given by (3.1). The RDS$^{(1)}$ value at the position j of the nth codeword $\underline{x}_n$ is given as:

$$RDS_{nN}^{(1)} = RDS_{(n-1)N}^{(1)} + rds_n^{(1)}. \tag{4.18}$$

For a multimode code, assume that during the nth encoding interval, the source word $\underline{y}_n$ has a binary pattern of $\underline{y}_a$ which is mapped into the codeword selection set C_a as shown in Figure 4.3.

A. MRDS(1) criterion

As described in Section 4.1, with dc^1-free codes the $MRDS^{(1)}$ selection criterion ensures that alternative codewords with opposite disparity in a selection set are appropriately selected to generate a dc^1-free coded sequence. This criterion can also be employed in construction of dc^1-free multimode codes [79], [87], and is used for dc control in current DVD systems [77].

From (4.18), during the nth encoding interval, given $RDS^{(1)}_{(n-1)N}$, the $MRDS^{(1)}$ criterion ensures selection of the codeword in C_a that results in the minimum absolute $RDS^{(1)}_{nN}$. Power spectra of GS 7/8, 14/16, and 21/24 dc^1-free multimode codes using $MRDS^{(1)}$ selection criterion are illustrated in Figures 4.5-4.8 where the solid curves represent the exact PSDs and the dashed curves represent the approximate PSDs obtained by evaluating the corresponding LFSWs and using expression (3.63). These figures show that given the code rate, the low-frequency characteristics (LFSWs) of dc^1-free multimode codes using the $MRDS^{(1)}$ criterion are not automatically improved with an increase of codeword length, or equivalently an increase of coding complexity. A similar observation regarding sum variance has also been made in [21]. Alternatively, it has been shown [21] that given the codeword length, the encoder efficiency defined in (3.53) decreases with the increase of the number of redundant bits.

Since the $MRDS^{(1)}$ selection criterion for both dc^1-free bimode and multimode codes is primarily dedicated to simple implementation of high-rate dc^1-free codes, it is possible to develop other selection criteria for high-rate dc^1-free multimode codes with better performance.

B. MSW(1) criterion

The sum variance performance metric [62] has been discussed in Section 3.2. For large rejection of low-frequency components, due to Justesen's well-known relation given in (3.47), it is desirable that the value of sum variance is small. Let $\{RDS^{(1)}\}$ be the $RDS^{(1)}$ sequence corresponding to the coded sequence $\{x\}$. From (2.7), it is straightforward to show that:

$$RDS^{(1)}_l = x_1 + x_2 + \cdots + x_l. \tag{4.19}$$

From its definition, sum variance can be expression as: [11]

$$\sigma^2_{RDS^{(1)}} = E\left[\left(RDS^{(1)}_l\right)^2\right] \tag{4.20}$$

---

[11] Assume that the mean value of $RDS^{(1)}$ is zero [62].

which implies that all $\text{RDS}_l^{(1)}$ values in $\{\mathbf{RDS}^{(1)}\}$ contribute to the statistical parameter $\sigma^2_{\text{RDS}^{(1)}}$. Clearly then, to reduce the value of sum variance of a coded sequence, consideration of only minimizing the absolute value of $\text{RDS}^{(1)}$ at the end of each codeword is not enough. During selection of codewords in a dc^1-free multimode encoder, all $\text{RDS}_l^{(1)}$ values within a codeword should to be taken into account resulting in another criterion, the minimum squared weight selection criterion [21]. In this thesis, this criterion is denoted as $\text{MSW}^{(1)}$.

The squared weight of the codeword $\underline{x}_n$, w_n^2, is defined to be the sum of squared $\text{RDS}^{(1)}$ at each bit position of this codeword, i.e.:

$$w_n^2 = \sum_{j=1}^{N} \left(\text{RDS}_{n,j}^{(1)} \right)^2. \qquad (4.21)$$

By use of the $\text{MSW}^{(1)}$ criterion, the codeword with minimum w_n^2 value is selected in a selection set. Given other coding parameters, it has been shown that the performance of dc^1-free multimode codes using the $\text{MSW}^{(1)}$ criterion is superior to that of dc^1-free multimode codes using the $\text{MRDS}^{(1)}$ criterion [21].

In Figures 4.5-4.8, power spectra of alternative GS dc^1-free multimode codes generated with the $\text{MSW}^{(1)}$ and $\text{MRDS}^{(1)}$ criteria are compared. The figures illustrate that both scrambling polynomials and selection criterion affect PSD of a code. In addition, it is shown that for a given code rate, an increase in the number of redundant bits can result in consistent improvement of LFSW when using the $\text{MSW}^{(1)}$ criterion. This agrees with the result in [21] where encoder efficiency is used for performance assessment.
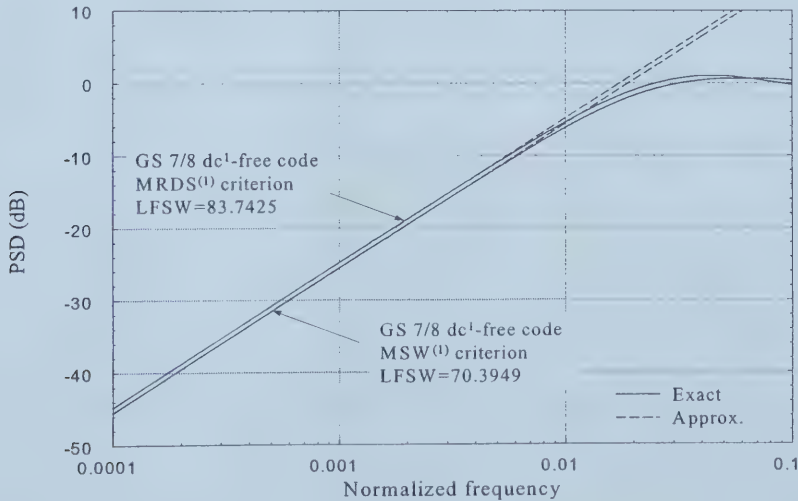
Figure 4.5 Power spectra of GS 7/8 dc^1-free multimode codes, $d(x) = x+1$.
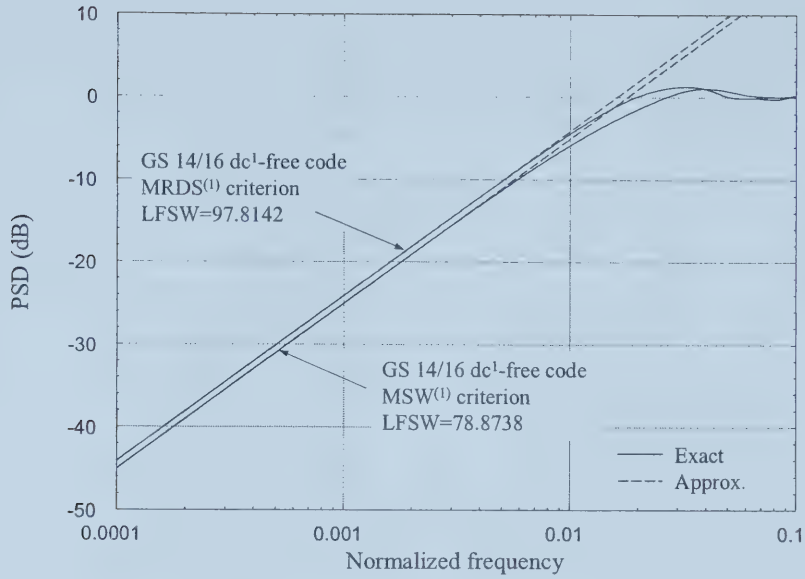
Figure 4.6 Power spectra of GS 14/16 dc^1-free multimode codes,
$d(x) = x+1$.
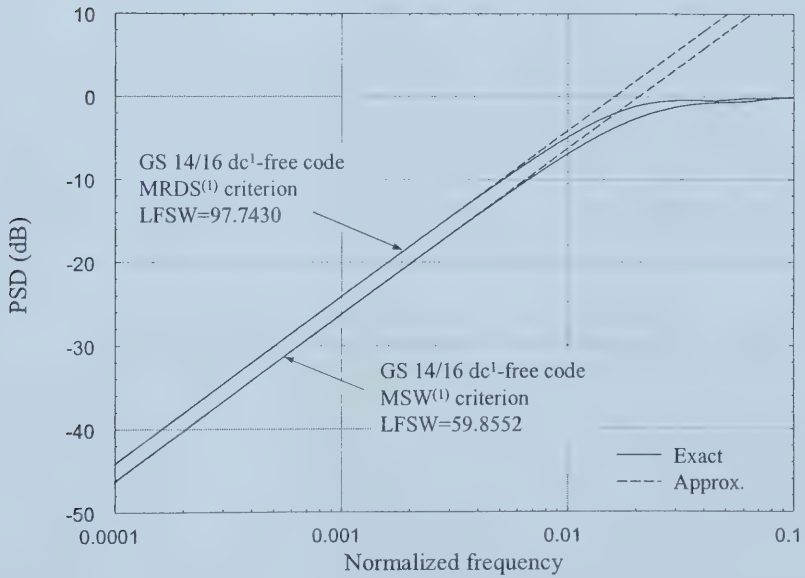
Figure 4.7 Power spectra of GS 14/16 dc^1-free multimode codes,
$d(x) = x^2+1$.
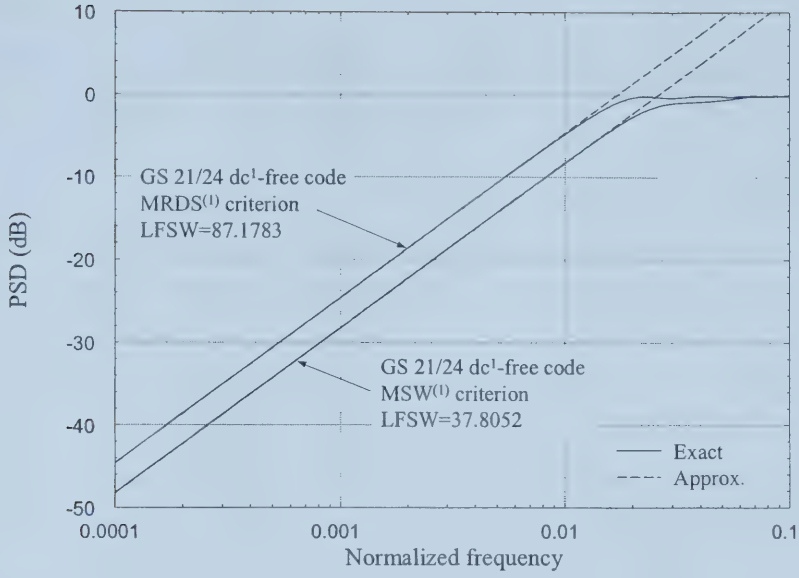
Figure 4.8 Power spectra of GS 21/24 dc^1-free multimode codes, $d(x) = x^2+1$.

Other selection criteria including the *modified MRDS$^{(1)}$ criterion* and the *minimum threshold overrun criterion*, have also been proposed in [21], however it is also shown there that these two criteria result in inferior suppression of low-frequency components when compared to the MSW$^{(1)}$ criterion. Still other heuristic selection criteria have recently been proposed for dc^1-free codes for suppression of low-frequency components by controlling the variation of RDS$^{(1)}$ or RDS$^{(2)}$ of the coded sequences [19], [31], [76]. A new general selection criterion based on the new performance metric LFSW introduced in Chapter 3 for arbitrarily HOSN codes is proposed in Chapter 5.

Chapter 5

EFFICIENT HIGH-ORDER SPECTRAL-NULL MULTI-MODE CODES [12]

In this chapter, a new class of codes with high-order spectral nulls at zero frequency is introduced. The codes, called high-order spectral-null multimode codes, are constructed through the guided scrambling coding technique and result in efficiently coded sequences with good low-frequency characteristics. To develop these codes, characterization of block coded sequences is first considered, and then a high-rate state-dependent encoding scheme for arbitrarily high-order spectral-null codes is proposed. It is shown that the low-frequency spectrum-weight of state-dependent encoded sequences generated through concatenation of fixed-length words equals the LFSW of the corresponding state-independent encoded sequences consisting of variable-length words. An expression to evaluate the LFSW of state-independent variable-length encoded sequences is derived, and gives insight regarding the low-frequency content of encoded sequences with spectral nulls at zero frequency. Two selection criteria for multimode codes are proposed and evaluated. Performance results that demonstrate the advantages of the new codes are presented.

In Chapter 3, it was shown that power spectra of high-order spectral-null codes defined in Chapter 2 demonstrate advantages in suppression of low-frequency components. As discussed in Chapter 3, block encoding/decoding provides recognized mappings between source words and codewords. A block encoder may be either state-independent or state-dependent. However, to limit error propagation during decoding, a decoder should be state-independent. Denote the set of length-N dc^K-free words as $\varphi(N,K)$, and denote a dc^K-free code generated through concatenation of length-N words to be from the set $\Phi(N,K)$.

Recall from Chapter 4 that codes can be categorized as monomode, bimode, or multimode codes in which each source word is uniquely represented by one codeword, two codewords, or a selection set containing more than two codewords respectively. State-independent encoding of block dc^K-free codes can be performed through free concatenation of independent dc^K-free words [23], [27]. Since every source word maps to only one codeword, such codes are called dc^K-free monomode codes (also called Kth-order zero-disparity codes) which are denoted to be from set $\Phi_{mo}(N,K)$. Several properties of dc^K-free monomode codes have been discussed in

---

[12] A version of this chapter has been submitted in part for publication. Y. Xin and I. J. Fair. *IEEE Trans. Commun.*

Section 2.3 and power spectra of this kind of codes were considered in Section 3.1.2. Some efficient encoding algorithms for generation of dc^K-free monomode codes $(K = 2 \text{ or } 3)$ have been proposed in [22], [27], [28], [30]. It has been shown, however, that as the order of the spectral null increases, the number of dc^K-free words of practical length decreases significantly [23], resulting in low rate dc^K-free monomode codes which limits their application in practical systems. To maintain efficiency in dc^K-free monomode coding, the word length must be increased [28], [30]. However, this leads to an increase in the complexity of encoding/decoding and an increase in error propagation, as well as poorer suppression of low-frequency content [22]. This was demonstrated in Figures 3.5-3.8.

Alternatively, bimode coding, a state-dependent encoding technique, has been applied to the construction of more rate-efficient dc^2-free codes [22], [26]. In Section 5.1.2, bimode coding is generalized for generation of dc^K-free codes ($K > 1$), and it is shown that the code rate can be significantly improved through concatenation of dependent dc^{K-1}-free words. However, for a given codeword length, it has been shown that the spectral suppression ability of bimode codes is weaker than that of monomode codes [22], [26].

The HOSN multimode codes introduced in Section 5.1.3 can be employed to improve the spectrum performance with little loss of rate efficiency compared to bimode codes of the same codeword length. Since spectrum evaluation plays a key role in the design and analysis of HOSN codes, this topic is discussed in Section 5.2 prior to the construction of HOSN multimode codes in Section 5.3.

Spectral suppression of dc^1-free codes is generally evaluated by the width of the spectral notch at low frequencies through sum variance [62]. Low-frequency spectrum-weight introduced in Section 3.3 is a new performance metric for codes with an arbitrarily high-order spectral null at dc. In contrast to sum variance, LFSW assesses the depth of spectral null around dc. In Section 5.2, an alternative, equivalent state-independent variable-length encoding model is introduced in place of the state-dependent fixed-length encoding description of the encoder. Using this model, a closed-from expression for the LFSW of these dc^K-free codes is derived. The expression for LFSW of the equivalent codes gives insight into the construction of good dc^K-free codes, as explained in Section 5.3.

Section 5.3 constructs efficient dc^K-free multimode codes ($K > 1$) through guided scrambling [79]. Scrambling polynomials are discussed and given. New methods of selecting codewords from quotient selection sets which result in generation of a Kth-order spectral-null sequence are introduced. Code rates of GS dc^K-free multimode codes are listed.

In Section 5.4, power spectra of various GS dc^K-free multimode codes are presented and compared with spectra of dc^K-free monomode codes. These spectra indicate that the proposed GS dc^K-free multimode codes can yield large rejection of low-frequency content and high code rate compared to monomode codes with the same order of spectral null and word length.

5.1 Rate-Efficient HOSN Codes

5.1.1 Characterization of Block Coded Dc^K-Free Sequences

Characterization of symbol sequences with a Kth-order spectral-null at dc has been developed in [24], [25]. In order to construct an efficient dc^K-free block code, further characterization of block coded sequences with a high-order spectral null at dc needs to be considered.

Let $\{x_l\} = x_1, \cdots, x_L$ be a symbol sequence of length L over the bipolar alphabet $\{-1, +1\}$. High-order RDS is defined in (2.7). Assume that the initial values of $\mathrm{RDS}^{(r)}$, $\mathrm{RDS}_0^{(r)}$ $(r = 1, \cdots, K)$, of the sequence are zero. Let the symbol sequence $\{x_l\}$ be formed from a word sequence $\{\underline{x}_n\}$ through concatenation of words, where $\underline{x}_n = (x_{n,1}, \cdots, x_{n,N}) \in \{-1, +1\}^N$ is a length-N word at the nth coding interval $(n \geq 1)$; then from definition (3.1), $x_{(n-1)N+j} = x_{n,j}$, $j \in \{1, \cdots, N\}$. Let $\mathrm{rds}_n^{(K)}$ satisfying (2.7) be the Kth-order RDS evaluated over the duration of the codeword $\underline{x}_n$. Also define the rth-order moment of the word $\underline{x}_n$ and the rth-order moment of the sequence $\{x_l\}$ respectively as:

$$u_n^{(r)} = \sum_{j=1}^{N} j^r x_{n,j} \text{ and } U_L^{(r)} = \sum_{l=1}^{L} l^r x_l, \ r \geq 0. \tag{5.1}$$

From definition (2.15), for a dc^K-free word $\underline{x}_n$, there is:

$$u_n^{(r)} = \sum_{j=1}^{N} j^r x_{n,j} = 0, \text{ for } r = 0, 1, \cdots, K-1. \tag{5.2}$$

Lemma 5.1: If the first K moments of every codeword are zero, the first K moments of the sequence constructed by concatenating these codewords are also zero, and the Kth-order moment of the sequence equals the sum of the Kth-order moments of the codewords.

Proof: Consider a length-L sequence consisting of H length-N codewords such that $L = H \cdot N$. Assume that the first K moments of the nth codeword $(1 \leq n \leq H)$ are zero, i.e. $\sum_{j=1}^{N} j^r x_{n,j} = 0$, $r = 0, 1, \cdots, K-1$. Then,

$$((n-1)N+1)^r x_{(n-1)N+1} + \cdots + ((n-1)N+N)^r x_{nN} = \sum_{j=1}^{N} j^r x_{n,j} = 0$$

$$((n-1)N+1)^K x_{(n-1)N+1} + \cdots + ((n-1)N+N)^K x_{nN} = \sum_{j=1}^{N} j^K x_{n,j} = u_n^{(K)},$$

where $u_n^{(K)}$ and $x_{(n-1)N+j} = x_{n,j}$ are as defined previously. Therefore, the moments of the sequence are:

$$U_L^{(r)} = \sum_{l=1}^{L} l^r x_l = 0, \ r = 0, 1, \cdots, K-1 \text{ and } U_L^{(K)} = \sum_{l=1}^{L} l^K x_l = \sum_{n=1}^{H} u_n^{(K)} . \qquad \square$$

According to the discussion in [24, pp. 833-834] and the lemma above, it can be shown that in a sequence generated through concatenation of dc^{K-1}-free words, at the end of each codeword, the values of $RDS^{(r)}$ $(r = 1, \cdots, K-1)$ are zero. Also, at the end of the nth codeword, the value of $RDS^{(r)}$ equals the sum of the $rds^{(K)}$ values of the first n codewords, and the $RDS_{nN}^{(K)}$ value equals the product of $U_{nN}^{(K-1)}$ and a constant. If the values of $RDS^{(K)}$ or $U^{(K-1)}$ at the end of each dc^{K-1}-free codeword are bounded, the sequence constructed by concatenating those codewords must have a Kth-order spectral null at dc [26].

5.1.2   Bimode Codes

As indicated above, Kth-order spectral-null codes $(K > 1)$ can be generated through appropriate concatenation of $(K-1)$th-order spectral-null words. Let dc^{K-1}-free words with positive, negative, and zero $rds^{(K)}$ values be from sets $\varphi_+(N, K-1)$, $\varphi_-(N, K-1)$, and $\varphi(N, K)$ [13] respectively. As with dc^1-free bimode codes [16], there are two methods of mapping source words to codewords in dc^K-free bimode codes:

Method 1: Each source word has two coded representations which are not both from the set $\varphi_+(N, K-1)$ or both from the set $\varphi_-(N, K-1)$. Denote such dc^K-free codes to be from set $\Phi_{bi1}(N, K)$;

Method 2: A source word is represented either with two different dc^{K-1}-free words, one from the set $\varphi_+(N, K-1)$ and the other from the set $\varphi_-(N, K-1)$, or is represented by one dc^K-free word from the set $\varphi(N, K)$. Denote such dc^K-free codes to be from set $\Phi_{bi2}(N, K)$.

In both methods of dc^K-free bimode encoding, codewords are selected to minimize the absolute $RDS^{(K)}$ value at the end of the codeword.

---

[13] Note that a $dc^{(K-1)}$-free word with an $rds^{(K)}$ value equal to zero is a dc^K-free word.

Let the size (the number of available codewords) of a set of codes ψ be denoted $|\psi|$. The theoretical rate of codes in $\Phi(N,K)$ is $(1/N)\log_2|\Phi(N,K)|$. The asymptotic bound on the size of codes in the set $\Phi_{mo}(N,K)$ has recently been developed [29]. It is straightforward to show that:

$$|\Phi_{bi1}(N,K)| = 1/2|\Phi_{mo}(N,K-1)|. \tag{5.3}$$

To demonstrate the efficiency of bimode codes compared to monomode codes, Table 5.1 gives the theoretical code rates[14] for codes in the set of $\Phi_{mo}(N,K)$ (columns labeled Mo), $\Phi_{bi1}(N,K)$ (columns labeled Bi1), and $\Phi_{bi2}(N,K)$ (columns labeled Bi2) for $N \le 32$. Note that for some cases of $N > 32$ and $K \ge 3$, theoretical code rates of monomode and bimode codes can be obtained from Tables 2.1 and 2.5. Table 5.1 also shows that for some word lengths and spectral null orders, bimode coding can generate dc^K-free codes which are not possible through monomode coding.

Table 5.1   Theoretical code rates for codes from $\Phi_{mo}(N,K)$, $\Phi_{bi1}(N,K)$, and $\Phi_{bi2}(N,K)$

K N	2			3			4		
	Mo	Bi1	Bi2	Mo	Bi1	Bi2	Mo	Bi1	Bi2
4	0.250	0.396	0.500	0	0	0	0	0	0
6	0	0.554	0.554*	0	0	0	0	0	0
8	0.375	0.641	0.661	0.125	0.250	0.290	0	0	0
10	0	0.698	0.698*	0	0	0	0	0	0
12	0.488	0.738	0.745	0.083	0.405	0.409	0	0	0
14	0	0.767	0.767*	0	0	0	0	0	0
16	0.565	0.791	0.794	0.238	0.502	0.505	0.063	0.175	0.188
18	0	0.809	0.809*	0	0	0	0	0	0
20	0.621	0.825	0.827	0.279	0.571	0.571	0	0.229	0.229*
22	0	0.838	0.838*	0	0	0	0	0	0
24	0.662	0.849	0.850	0.384	0.621	0.621	0.167	0.342	0.344
26	0	0.858	0.858*	0	0	0	0	0	0
28	0.695	0.866	0.867	0.411	0.659	0.660	0	0.375	0.375*
30	0	0.874	0.874*	0	0	0	0	0	0
32	0.721	0.880	0.881	0.472	0.690	0.690	0.196	0.440	0.440

Note: * denotes cases where the code rate of codes in $\Phi_{bi2}(N,K)$ is identical to that of codes in $\Phi_{bi1}(N,K)$.

In Figure 5.1, low-frequency spectra of selected third and fourth-order spectral-null codes are presented. In the lookup tables for generation of the dc^K-free bimode codes shown in this figure,

---

[14] calculated from the well-known result $|\Phi_{mo}(N,1)| = C_{N/2}^N$ and from the table in [23].

codewords were selected with absolute values of $u^{(K-1)}$ as small as possible. This figure and data in Table 5.1 demonstrate that for the same word length, bimode codes can realize the same order of spectral null as monomode codes but with higher code rate. It is also evident that, compared to mapping method 1, mapping method 2 may either increase the code rate or improve the spectral performance of dc^K-free bimode codes. However, Figure 5.1 also shows that the higher efficiency of dc^K-free bimode codes is obtained at the expense of spectral performance at low frequencies compared to the same order spectral-null monomode codes with the same word length.
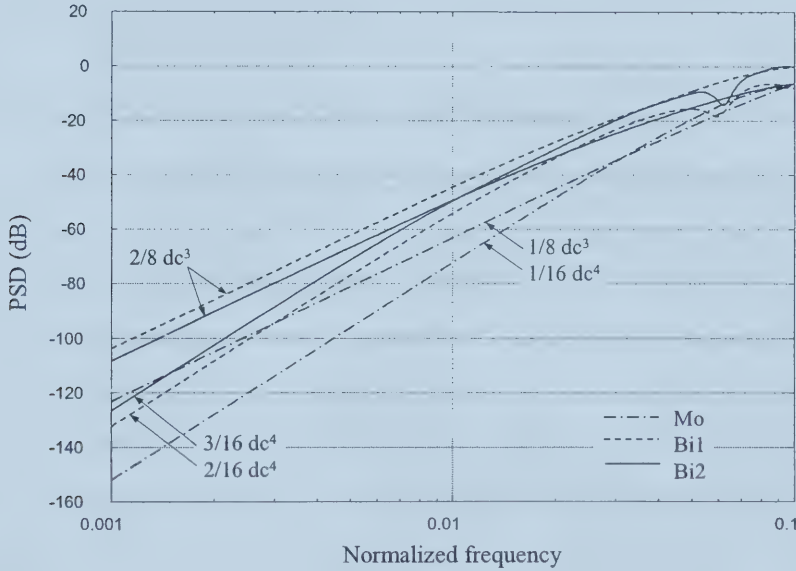
Figure 5.1 Power spectra of dc^K-free monomode codes and bimode codes $(K = 3, 4)$.

5.1.3 Multimode Codes

To implement a rate-efficient dc^K-free code with a deep spectral null at low frequencies, dc^K-free multimode coding is considered. Similar to the dc^K-free bimode codes above, construction of dc^K-free multimode codes in this thesis is also based on the concatenation of dc^{K-1}-free words. To construct a coded sequence with a Kth-order spectral-null at dc, it is required that the $RDS^{(K)}$ values at the end of each dc^{K-1}-free codeword be bounded. To ensure these bounds, in this thesis each source word is mapped into a codeword selection set that includes at least one pair of complementary words with a $(K - 1)$th-order of spectral null at dc. Multiple pairs of complementary dc^{K-1}-free words, if they exist in the selection set, provide an opportunity for a multimode encoder to choose the "best" codeword in each particular encoding interval. Note

that like-order moments and RDS values of complementary words have equal magnitude but opposite sign. It can be shown that given the word length N, the code rate of such dc^K-free multimode codes is upper bounded by the theoretical code rates for codes in $\Phi_{bi1}(N, K)$ shown in Table 5.1.

The spectral suppression ability of a dc^K-free multimode code is determined by both the mapping between source words and codewords, and the codeword selection criteria. Spectral suppression of a dc^K-free code is indicated by its LFSW, which is desired to be as small as possible. In the next section, an expression for the LFSW of state-independent variable-length dc^K-free codes is derived for the purpose of developing a good quotient selection criterion for GS codes. This criterion is described in Section 5.3.

5.2 LFSW of Variable-Length HOSN Codes

To permit performance assessment and design of dc^K-free codes, low-frequency characteristics of dc^K-free codes need to be determined. In Section 3.3, it was shown that the asymptotic continuous low-frequency spectral components of a sequence $\{x_l\}$ with a Kth-order spectral-null at dc $(K \geq 1)$ are exclusively determined by the order K and the low-frequency spectrum-weight, and that the LFSW equals the zero-frequency value of the continuous power spectrum of the corresponding sequence of Kth-order RDS values, $\{RDS_l^{(K)}\}$. At low frequencies, the approximate continuous PSD of $\{x_l\}$, $H_x(\omega)$, is related to the LFSW, $H_{RDS^{(K)}}(0)$, through expression (3.63). Given the order K, the smaller the value of LFSW, the greater the suppression of low-frequency components of the dc^K-free code. In this thesis the right side of (3.63) is referred to as the asymptotic PSD.

Calculation of the LFSW of a fixed-length dc^K-free code $(K \geq 1)$ has been introduced in Section 3.3.2. Based on the work in [34] regarding evaluation of spectra of block coded sequences, the value of $H_{RDS^{(K)}}(0)$ for fixed-length dc^K-free sequences generated through state-dependent encoding can be obtained with the modification that the output of the finite-state sequential machine is the sequence of $RDS^{(K)}$ values rather than the sequence of symbols. Furthermore, a closed-form expression for $H_{RDS^{(K)}}(0)$ of fixed-length dc^K-free sequences generated through state-independent encoding, which provides insight into how to obtain a small LFSW value, has been derived in Section 3.3.3 and given by Theorem 3.5.

As described in Section 5.1, rate-efficient dc^K-free bimode and multimode codes are generated through state-dependent encoding. As discussed in Section 3.3.2, the LFSW of a dc^K-free code generated through state-dependent encoding is obtained through numerical analysis based on complete knowledge of state-transition information. However, unlike the state-independent case, during construction of a fixed-length dc^K-free bimode or multimode code, it is not clear how to obtain a small LFSW value. Therefore, there is motivation to consider an alternative representation of the state-dependent dc^K-free length-N code which has the same power spectrum performance and state-independent characteristics, and to develop an intuitive relationship between the LFSW value and other characteristics of the coded sequence.

As described in Section 2.2.2, a symbol sequence $\{x_i\}$ with Kth-order spectral-null constraints at dc generated through concatenation of words can be characterized by an irreducible finite-state transition diagram [24]-[26] G with an associated Markov chain. For a dc^K-free block code, a FSTD G^N, an Nth extension of the corresponding G, can be used to describe the dc^K-free fixed-length encoder. Let permitted state sequences of the chain in G^N be denoted $\{s_n\}$, where each state in $\{s_n\}$ is represented by finite values of RDS$^{(r)}$ $(r=1,\cdots,K)$ at the end of codewords. The word sequence $\{\underline{x}_n\}$ is generated by G^N. Recall that $x_{(n-1)N+j} = x_{n,j}$, and let RDS$_l^{(r)}$ and $\underline{\text{RDS}}_n^{(r)}$ be related to x_l and $\underline{x}_n$ through (2.7) respectively. Since G^N is also irreducible, there must exist a cycle of states satisfying $\{s_0, s_1, \cdots, s_\upsilon = s_0\}$ [24]-[26], where υ is the length of the cycle. As in Section 5.1, let the initial state of each cycle be given when RDS$_0^{(r)} = 0$, $r=1,\cdots,K$. Therefore, RDS$^{(r)}$ values $(r=1,\cdots,K)$ at the end of a length-υN symbol sequence $\{x_1, x_2, \cdots, x_{\upsilon N}\}$ are all zero; equivalently, the rth-order moment of the sequence, $U_{\upsilon N}^{(r)}$, for $r=0,\cdots,K-1$ are all zero [24] implying that this symbol sequence is a dc^K-free word of length υN [27]. Since the length of the cycles will vary during encoding, the dc^K-free sequence constructed through concatenation of state-dependent fixed-length words can be viewed as a sequence composed of dc^K-free words with variable lengths, where these lengths are divisible by N. Because each cycle of states is mutually independent, the generated dc^K-free words associated with such cycles are also mutually independent. The resulting sequences can be interpreted as state-independent variable-length dc^K-free sequences.

The power spectrum of a symbol sequence composed of mutually independent variable-length words was developed in [70]. Below, the LFSW of the new state-independent variable-length dc^K-free sequences is calculated by considering the related state-independent variable-length RDS$^{(K)}$ sequences.

Assume that $\mathcal{R}$ is an infinite set of variable-length $\mathrm{RDS}^{(K)}$-valued words with lengths from an infinite set of word lengths $\mathcal{L}$ in which all elements are positive integers divisible by N. Let $\mathbf{RDS}_m^{(K)}$ be a vector of dimension L_m containing the $\mathrm{RDS}^{(K)}$ values of the mth cycle, and let $\{\mathbf{RDS}_m^{(K)}, \mathbf{RDS}_m^{(K)} \in \mathcal{R}\}$ be a variable-length $\mathrm{RDS}^{(K)}$-valued word sequence; $\{L_m, L_m \in \mathcal{L}\}$ is the corresponding sequence of variable word lengths. Denote the probabilities of length-λ cycles as $p_\lambda = \mathrm{Pr}\{L_m = \lambda\}$, $\lambda \in \mathcal{L}$. The mean word length is then: $\bar{\lambda} = \sum_\lambda \lambda p_\lambda$. To remain consistent with the notation of [70], let $\Gamma^{-1} = \bar{\lambda}$.

Let $\mathcal{R}_\lambda = \{\rho_\lambda(1), \rho_\lambda(2), \cdots, \rho_\lambda(\eta_\lambda)\}$ be a set of η_λ $\mathrm{RDS}^{(K)}$-valued words with length λ where $\rho_\lambda(i) = \left[\mathrm{RDS}_1^{(K)}(i), \mathrm{RDS}_2^{(K)}(i), \cdots, \mathrm{RDS}_\lambda^{(K)}(i) \right]$, $i = 1, \cdots, \eta_\lambda$. Assuming that the source sequence is composed of independent and equiprobable symbols, all η_λ words of length λ have the same probability. From (11) in [70], $\boldsymbol{m}_\lambda$, the mean vector of $\mathrm{RDS}^{(K)}$-valued words of length λ, is:

$$\boldsymbol{m}_\lambda = E\{\mathbf{RDS}_m^{(K)}; L_m = \lambda\} = \frac{1}{\eta_\lambda} p_\lambda \sum_{i=1}^{\eta_\lambda} \rho_\lambda(i). \tag{5.4}$$

Theorem 5.1: The LFSW of state-independent variable-length dc^K-free sequences is:

$$H_{\mathrm{RDS}^{(K)}}(0) = \Gamma\, E\left\{\left(\mathrm{RDS}_m^{(K+1)}\right)^2\right\} + \Gamma \sum_{\lambda,\mu} \left[2X(1) + N\Gamma(1 - \lambda' - \mu')\right] V_\lambda \boldsymbol{m}_\lambda \boldsymbol{m}_\mu V_\mu', \tag{5.5}$$

where V_λ is all-ones vector of dimension λ, V_λ' is its transpose, and:

$$\lambda' = \lambda N^{-1}, \ \mu' = \mu N^{-1}, \ \lambda, \mu \in \mathcal{L},$$

$$E\left\{\left(\mathrm{RDS}_m^{(K+1)}\right)^2\right\} = \sum_\lambda \frac{1}{\eta_\lambda} p_\lambda \sum_{i=1}^{\eta_\lambda} \left(\mathrm{RDS}_\lambda^{(K+1)}(i)\right)^2,$$

$$\mathrm{RDS}_\lambda^{(K+1)}(i) = \sum_{j=1}^{\lambda} \mathrm{RDS}_j^{(K)}(i),$$

$$X(1) = \frac{N\Gamma \sum_\lambda p_\lambda (1/2)\lambda'(\lambda' - 1)}{\sum_\lambda p_\lambda \lambda'}.$$

Proof: Let the notations be as defined above. The proof of the theorem is based on the work in [70] which is reviewed in Section 3.1.3, however calculation of LFSW is simplified because LFSW is the continuous spectral component of the $\mathrm{RDS}^{(K)}$ sequence evaluated only at zero-frequency, therefore the notations $z = e^{i\omega T_b}$ and $V_\lambda(z)$ in [70] reduce to 1 and V_λ respectively, where V_λ is the all-ones vector of dimension λ. Furthermore,

$$X(1) = \lim_{z \to 1} X(z) = \frac{NT \sum_{\lambda} p_{\lambda}(1/2)\lambda'(\lambda'-1)}{\sum_{\lambda} p_{\lambda}\lambda'} \tag{5.6}$$

In this application, the greatest common divisor of cycle lengths Λ defined in [70] equals the word length N. Let an $\eta_{\lambda} \times \lambda$ matrix be $\boldsymbol{\rho}_{\lambda} = [\boldsymbol{\rho}_{\lambda}(1),\cdots,\boldsymbol{\rho}_{\lambda}(\eta_{\lambda})]'$ where the prime denotes transposition, and let $\boldsymbol{\rho}_{\lambda}(\varsigma_{\lambda})$ $(1 \le \varsigma_{\lambda} \le \eta_{\lambda})$ be the λ-dimension vector of $\text{RDS}^{(K)}$-values of the ς_{λ}th word within the set of words of length λ. From (43) of [70], the autocorrelation matrices of $\text{RDS}^{(K)}$-valued words of length λ with zero separation can be expressed as:

$$\boldsymbol{R}^{\lambda\lambda}_{\text{RDS}^{(K)}}(0) = \frac{1}{\eta_{\lambda}} p_{\lambda}\boldsymbol{\rho}'_{\lambda}\boldsymbol{\rho}_{\lambda}. \tag{5.7}$$

Let $\boldsymbol{H}^{\lambda\mu}_{\text{RDS}^{(K)}}(0)$ be the continuous part of the spectral density of the variable-length $\text{RDS}^{(K)}$-valued word sequence at zero frequency, and let the $H_{\text{RDS}^{(K)}}(0)$ be the corresponding value of LFSW. From Theorem 3.2 and expression (3.46), there is:

$$H_{\text{RDS}^{(K)}}(0) = \Gamma \sum_{\lambda,\mu} V_{\lambda} \boldsymbol{H}^{\lambda\mu}_{\text{RDS}^{(K)}}(0) V'_{\mu} \tag{5.8}$$

$$\boldsymbol{H}^{\lambda\mu}_{\text{RDS}^{(K)}}(0) = \boldsymbol{R}^{\lambda\lambda}_{\text{RDS}^{(K)}}(0) + \left[2X(1) + NT(1-\lambda'-\mu')\right]\boldsymbol{m}_{\lambda}\boldsymbol{m}_{\mu} \tag{5.9}$$

where $\lambda' = \lambda N^{-1}$, $\mu' = \mu N^{-1}$. Replacing (5.4), (5.6), (5.7), and (5.9) into (5.8) completes the proof. $\qquad\square$

Remark 5.1: From this Theorem it can be concluded that if the summation in (5.5) is zero, the LFSW value is determined only by the mean values of the word length and the mean squared $\text{RDS}^{(K+1)}$ values at the end of the variable-length words. With practical codes it is expected that the mean of the squared $\text{RDS}^{(K+1)}$ values at the end of variable-length words will be small and the mean word length will be large.

Remark 5.2: Since the variable-length sequences considered above can have infinite word length, the expression for LFSW of state-independent variable-length dc^K-free sequences given in the Theorem cannot be applied to an exact calculation of LFSW. However, this analysis is applicable to simulation of LFSW since probabilities of variable-length words with word lengths λ decrease as $O(2^{-\lambda})$ while the values of squared $\text{RDS}^{(K+1)}$ increase only in proportion to $\lambda^{2(K+1)}$ with large λ. This relationship results from that fact that a length-λ codeword is mapped to a length-λR source word, and the values of squared $\text{RDS}^{(K+1)}$ is in proportion to the

squared value of Kth-moment of this word $u^{(K)}$ [24] and $\sum_{j=1}^{\lambda} j^K = O(\lambda^{K+1})$ [58, p. 1]. Simulation test results show that with long simulation sequences, the simulated LFSW value is very close to the exact value.

In the next section, the analysis above is applied to the design of dc^K-free multimode codes that result in better spectrum performance and higher rate-efficiency than dc^K-free monomode codes.

5.3 Encoding of GS HOSN Multimode Codes

The guided scrambling multimode coding technique has been discussed in Section 4.2. Since GS decoding is trivial, only GS encoding is considered in this section.

5.3.1 Scrambling Polynomials for HOSN Multimode Codes

In a guided scrambling encoder, scrambling polynomials determine the mapping between source words and quotient selection sets. In the application of HOSN multimode codes introduced in Section 5.1, the scrambling polynomial must be chosen such that there exists at least one complementary dc^{K-1}-free word pair in every quotient selection set. In Theorem 4.1 it has been shown that base polynomials of even weight are necessary and sufficient to generate complementary quotients in quotient selection sets. Conditions regarding the form of scrambling polynomials that ensure the existence of a dc^{K-1}-free word $(K>1)$ in every selection set remains an open question. In this thesis, base scrambling polynomials for high-order spectral-null GS codes have been found through computer search.

Table 5.2 lists base polynomials for a number of GS codes that guarantee that every selection set contains at least one pair of complementary dc^{K-1}-free words. In this table, M and N are as defined previously, R/R_T is the ratio of the rate of the highest rate GS code to the theoretical maximum code rate of the associated bimode code which is listed in Table 5.1 in columns labeled Bi1, and the scrambling polynomials are represented by their coefficients, from most to least significant coefficient. For instance, the scrambling polynomial $d(x) = x^5 + x^2 + x + 1$ is represented by $d = 100111$. Note that, given the codeword length, some of these codes are the highest rate Kth-order spectral-null codes reported to date in the literature, and alternatively, given the code rate, the word lengths of some of these new codes are very short compared to other recently published results. For example, for a code rate of about 0.5, word lengths of about 60 are reported for dc^2-free monomode codes [30] and dc^3-free monomode codes [28]. More

available base scrambling polynomials for GS HOSN multimode codes can be found in Appendix 3.

Table 5.2   Base scrambling polynomials for Kth-order spectral-null GS codes

K = 2			K = 3			K = 4		
M/N	R/R_T	Scrambling Polynomial	M/N	R/R_T	Scrambling Polynomial	M/N	R/R_T	Scrambling Polynomial
1/4	0.631	11						
2/6	0.602	11						
4/8	0.780	11011	2/8	1.000	1001			
5/10	0.716	101						
7/12	0.790	1001	4/12	0.823	110101			
8/14	0.745	101						
10/16	0.790	10001	6/16	0.747	1000111	2/16	0.714	1001
12/18	0.824	100111						
14/20	0.848	101011	9/20	0.788	11000101	4/20	0.873	1101100101
15/22	0.814	100111						
17/24	0.834	10111	12/24	0.805	100010101	6/24	0.731	10011111
19/26	0.852	100111						
21/28	0.866	1000111	15/28	0.813	100010011	8/28	0.762	100101111
23/30	0.877	1010011						
24/32	0.852	1000111	18/32	0.815	1000000111	11/32[**]	0.781[**]	10100111111

[**]Highest rate GS code found to date.

In the remainder of this section, new methods of selecting codewords from quotient selection sets are introduced.

5.3.2   Selection Criteria for HOSN Multimode Codes

A.   Primary Selection

As outlined in [79], the most straightforward approach for selecting a codeword in GS coding is to generate all 2^{N-M} quotients in the quotient selection set, examine each one for the desired criterion, and select the quotient that best meets the requirements. The task of examining potential codewords for dc^K-free codes can be simplified with the following observation. Note that scrambling polynomials of even weight generate quotient selection sets that consist of 2^{N-M-1} pairs of complementary quotients. It is straightforward to verify that quotients with a most significant coefficient equal to zero (one) are generated from augmented words whose most significant coefficient is also zero (one). Since complementary quotients have moments (or RDSs) of the same magnitude but opposite sign, examination of half of the quotients, those generated from augmented words whose most significant coefficient is zero (one), is sufficient to determine the suitability of all quotients in the set.

As defined in (5.1), let the rth-order moment of the encoded sequences of n codewords be $U_{nN}^{(r)}$ and the rth-order moment of the nth codeword be $u_n^{(r)}$. From Lemma 5.1, it can be shown that in sequences formed through concatenation of dc^{K-1}-free words, $U_{nN}^{(r)} = 0, r = 0, 1, \cdots, K-2$, and $U_{nN}^{(K-1)} = U_{(n-1)N}^{(K-1)} + u_n^{(K-1)}$. For generation of dc^K-free codes, it is sufficient that $U_{nN}^{(r)} = 0, r = 0, 1, \cdots, K-2$ and that $|U_{nN}^{(K-1)}|$ is upper bounded by a finite constant. These conditions, for GS codes, can be satisfied as follows:

- When $U_{(n-1)N}^{(K-1)} = 0$, if a quotient in the selection set is found to have $u_n^{(r)} = 0$, $r = 0, 1, \cdots, K-2$, it can be selected as the codeword.

- When $U_{(n-1)N}^{(K-1)} > 0$, if a quotient is found to have $u_n^{(r)} = 0, r = 0, 1, \cdots, K-2$, and $u_n^{(K-1)} \leq 0$, it can be selected. If a quotient is found to have $u_n^{(r)} = 0, r = 0, 1, \cdots, K-2$, and $u_n^{(K-1)} > 0$, its complement can be selected as the codeword.

- When $U_{(n-1)N}^{(K-1)} < 0$, if a quotient is found to have $u_n^{(r)} = 0, r = 0, 1, \cdots, K-2$, and $u_n^{(K-1)} \geq 0$, it can be selected. If the quotient is found to have $u_n^{(r)} = 0, r = 0, 1, \cdots, K-2$, and $u_n^{(K-1)} < 0$, its complement can be selected as the codeword.

Alternative selection criteria exist. For instance, since $RDS^{(K)}$ is a product of $u^{(K-1)}$ and a constant for a dc^{K-1}-free word [24], dc^K-free codes can also be formed by selecting codewords such that the $RDS^{(K)}$ at the end of each word is bounded. Since the scrambling polynomials listed in Table 5.2 and Appendix 3 guarantee that every selection set contains at least one pair of complementary dc^{K-1}-free words, a codeword satisfying the above selection criteria is guaranteed to exist in every quotient selection set. However, several quotients from the selection set could satisfy the criteria, necessitating secondary selection criteria. These criteria can be designed to optimize spectrum compression at low frequency or to simplify implementation. Below secondary selection alternatives are considered.

B. Secondary Selection Based on the $MSRDS^{(K+1)}$ Criterion

Secondary selection criteria for dc^K-free multimode codes $(K > 1)$ can be obtained by simply extending the established selection criteria for dc^1-free multimode codes discussed in Section 4.2.4, such as the $MRDS^{(1)}$ criterion, which minimizes the absolute values of word-end $RDS^{(1)}$ [79], [87], and the $MSW^{(1)}$ criterion, which minimizes the sum of the squared values of $RDS^{(1)}$ at each bit position within a codeword [21]. Denote these extensions as the $MRDS^{(K)}$ criterion,

which selects from the selection set the dc^{K-1}-free word with the minimal absolute value of the word-end $RDS^{(K)}$, and the $MSW^{(K)}$ criterion, which selects from the selection set the dc^{K-1}-free word with the minimal sum of the squared values of $RDS^{(K)}$ at each bit position within the codeword. However, regarding large suppression of low-frequency components, from Theorem 5.1, it can be understood that the $MRDS^{(K)}$ criterion, which focuses only on minimizing the $RDS^{(K)}$ values at the end of codewords, will not result in the best PSD performance; $RDS^{(K)}$ values at each bit position within the codewords should be considered. This has been verified for the case of dc^1-free multimode codes [21]. Furthermore, Theorem 5.1 also indicates that the $MSW^{(K)}$ criterion, which only takes into account individual $RDS^{(K)}$ values at each bit position, can be improved by considering correlation between $RDS^{(K)}$ values at different bit positions. Figures 5.2 and 5.3 demonstrate that for GS HOSN multimode codes, the $MSW^{(K)}$ criterion results in better performance than the $MRDS^{(K)}$ criterion when given other encoding parameters. These figures also show that given word length 20 and the order of spectral null, although multimode codes have higher code rate than monomode codes, low-frequency components in GS dc^2-free and dc^3-free multimode codes with the $MSW^{(K)}$ criterion are about 7 dB and 11 dB higher than those in dc^2-free and dc^3-free monomode codes respectively.
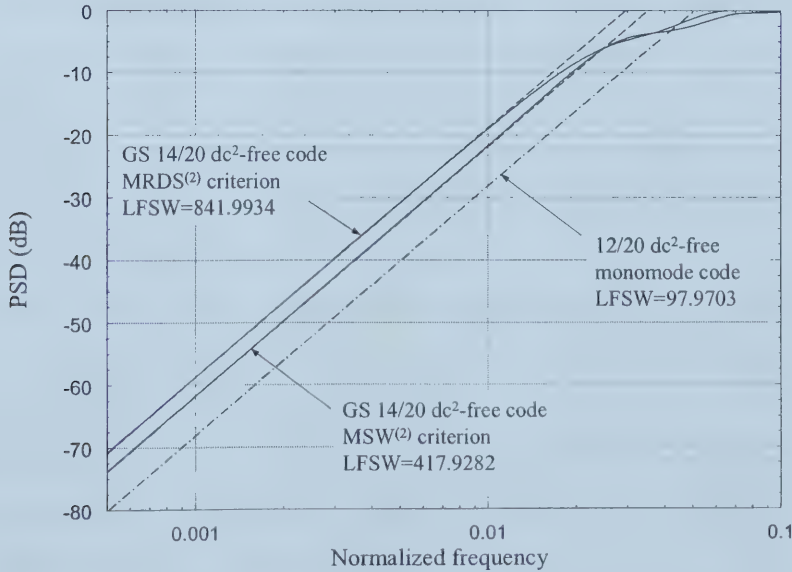
Figure 5.2 Comparison of power spectra of GS 14/20 dc^2-free multimode codes (d=101011) and 12/20 dc^2-free monomode code with the lowest LFSW.
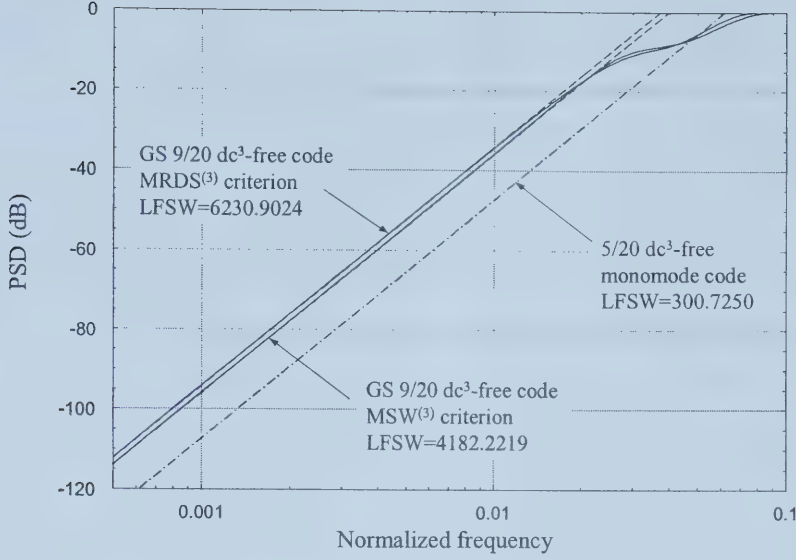
Figure 5.3 Comparison of power spectra of GS 9/20 dc³-free multimode codes (d=11000101) and 5/20 dc³-free monomode code with the lowest LFSW.

Based on the analysis in Section 5.2, to generate GS dc^K-free multimode codes ($K \geq 1$) in a simple manner with large suppression of low-frequency components, selecting the dc^{K-1}-free word[15] from the selection set with the minimal squared $RDS^{(K+1)}$ values at the end of the word is proposed. This selection criterion is called the $MSRDS^{(K+1)}$ criterion. From Theorem 5.1, in order to optimize the LFSW of coded sequences, all correlations between $RDS^{(K)}$ values at different bit positions within a variable-length sequence that is related to a state cycle and is possibly composed of multiple codewords of length N need to be considered. However, the infinite variable length of independent sequences of $RDS^{(K)}$ limits this consideration. While not optimal with respect to minimizing LFSW, this simple $MSRDS^{(K+1)}$ criterion guarantees that the squared $RDS^{(K+1)}$ values at the end of length-N dc^K-free codewords with the greatest probability in the coded sequence are minimal and avoids large delay during encoding. Note that quotients which satisfy the $MSRDS^{(K+1)}$ criterion also satisfy the primary selection criterion listed in the previous subsection, and that the $RDS^{(K+1)}$ value is reset to zero at the start of each cycle of states. Power spectra of some dc^K-free multimode codes based on the $MSRDS^{(K+1)}$ selection criterion will be considered in Section 5.4.

---

[15] Note that when $K = 1$ there is no constraint on the complementary words in the selection sets.

As discussed below, this selection mechanism will also ensure that the mean vectors of $RDS^{(K)}$-valued variable-length words are all-zero vectors. According to Remark 5.1, this will simplify our analysis. Characteristics of GS encoders and related encoded sequences have been investigated in [90]. Let complementary sets be as defined in Section 4.2 and in [90], where it is also shown that scrambling polynomials with even weight result in every quotient set being its own complement (indicated as Characteristic 4.2 in this thesis). In the present application, define complementary states to be states with the same $RDS^{(K)}$ magnitude but opposite polarity. Note that the state with zero $RDS^{(K)}$ value is its own complement; denote this state to be the central state. Similar to the discussion in [90], it can be shown that with the $MSRDS^{(K+1)}$ criterion, for any given source word, complementary quotients are ensured to be selected from complementary states, therefore the next states following codeword selection are also complementary. If the source words are stationary and memoryless, and if complementary quotients are selected with equal probability from the central state, complementary states will have equal stationary probabilities and equal transition probabilities to complementary states. This will result in equiprobable complementary codewords as well as equiprobable complementary words of variable length. Therefore, for each variable word length λ, m_λ is a zero-mean vector of $RDS^{(K)}$-valued words.

For even a moderate number of redundant bits $N - M$, however, generation and examination of half of the 2^{N-M} quotients in the selection set proves challenging in practical encoding. As an alternative, in the following subsection a serial selection mechanism is proposed in which only as many quotients in the selection set are generated as are required to find one that meets the primary criterion.

C. Serial Selection Mechanism

Consider Figure 5.4, which depicts all augmenting bit patterns that begin with a zero arranged on the perimeter of a circle. A similar circle can be drawn for augmenting bit patterns that begin with a one. Starting from any of the 2^{N-M-1} positions on one of these circles, quotients associated with these augmenting bit patterns can be generated and examined serially until an appropriate codeword is found. Movement on the circle can be in the following directions:

I)   in a clockwise direction;

II)   in a counterclockwise direction;

III) first in a clockwise direction, one position from the initial pattern, then in the counterclockwise direction, one position from the initial pattern, then in the clockwise direction two positions from the initial pattern, etc.;

IV) as in *III*), except with the first movement being in the counterclockwise direction;

V) simultaneously in both the clockwise and counterclockwise direction. This would require the use of two parallel scrambling and quotient examination circuits. When two appropriate quotients are found at the same time, consistently select the quotient generated in the clockwise (or counterclockwise) direction.

Clearly, other procedures for moving through the quotients can also be designed.

Initial augmenting
bit pattern

$0\ 0\ 1\ ...\ 1\ 1$

$0\ 1\ 0\ ...\ 0\ 0$
$0\ 1\ 0\ ...\ 0\ 1$

$0\ 0\ 0\ ...\ 0\ 1$
$0\ 0\ 0\ ...\ 0\ 0$

$0\ 1\ 1\ ...\ 1\ 1$
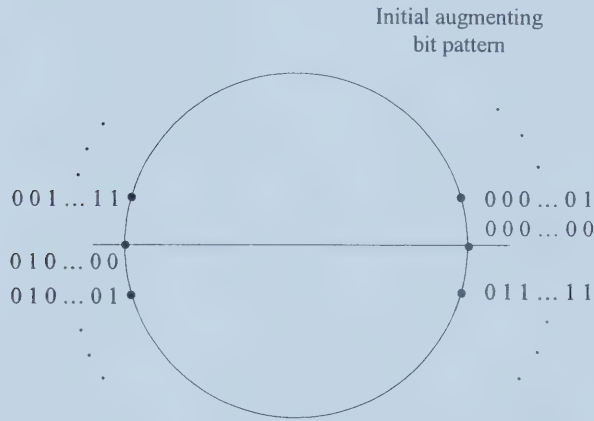
Figure 5.4 Circular configuration of augmenting bit patterns with most significant coefficient equal to zero. The initial augmenting bit pattern can be at any point on the circle.

Instead of examining all quotients in the selection set, the serial search process can be halted once a quotient with $u^{(r)} = 0, r = 0, 1, \cdots, K - 2$, has been found. This will result in codeword selection in the fewest number of quotient generating steps. This selection technique is called *bimode* because when all other code parameters are fixed, use of this criterion essentially results in a bimode, rather than a multimode, GS code.

To demonstrate the effectiveness of this serial selection mechanism, Table 5.3 presents results regarding the minimum average and corresponding maximum number of quotient generating steps, and the associated initial search positions[16] for a number of GS dc^2-free codes using the bimode stopping mechanism. In this table the scrambling polynomials used are from

---

[16] It is observed that the average number of quotient generating steps is dependent on the position in which the search commences.

the third column in Table 5.2. These results assume equiprobable source words, and therefore equiprobable quotients sets, and are given for the 2^{N-M-1} possible initial positions on the circle of augmenting bit patterns where the first bit is zero. In Table 5.3, the initial positions are indexed by the decimal representation of the associated augmenting bit pattern, e.g. $3 \leftrightarrow 0011$. The initial position index for Algorithm V, in which two quotients are generated and examined in each encoding interval, refers to the position that is furthest counterclockwise of the two patterns considered first.

Table 5.3 Characteristics of GS M/N dc^2-free codes using serial bimode quotient selection

	Alg.	4/8	7/12	10/16	14/20	17/24	21/28	24/32
Initial Index	I	3	7	59	36	121	51	38
	II	1	3	6	63	82	45	28
	III	3	4	7	35	17	48	31
	IV	2	4	7	35	16	47	30
	V	2	4	7	34	16	47	30
Minimum Average Number of Quotient Generating Steps	I	2.19	2.96	3.71	4.47	4.88	5.65	6.04
	II	2.38	3.06	3.60	4.42	5.08	5.60	5.66
	III	2.25	3.01	3.69	4.44	4.58	5.48	5.48
	IV	2.25	3.00	3.65	4.44	4.89	5.48	5.50
	V	1.38	1.77	2.10	2.48	2.70	2.99	2.97
Maximum Number of Quotient Generating Steps	I	5	9	17	31	29	60	115
	II	4	15	15	30	30	59	112
	III	4	9	17	32	34	60	67
	IV	4	8	17	32	33	59	67
	V	2	4	9	16	17	30	34

This data demonstrates that the minimum average number of quotient generating steps is surprisingly low for all codes considered. For example, in the 24/32 code, use of Algorithm V results in, on average, less than three quotient generating steps in each quotient selection set. Even though in general the corresponding maximum number of quotient generating steps is significantly larger than the average, the low average number of steps indicates that serial selection mechanisms are practical for GS codes with large quotient selection sets.

Note that the complexity of a GS dc^K-free encoder based on the new bimode selection criterion depends only linearly on the codeword length, order of spectral-null, and average number of quotients generated. Precomputation and storage of information are not required. Alternatively, enumerative coding has been shown to be impractical for HOSN codes [27]. Also note that the GS decoding process remains trivial, regardless of the code parameters, requiring only one shift register whose length depends on the degree of the scrambling polynomial.

5.4 Power Spectrum Results

In Figures 5.5-5.10 power spectra of various GS dc^K-free codes are presented. In Figure 5.5 the PSD performance of several GS 4/8 dc^2-free codes that use different scrambling polynomials and quotient selection criteria are compared. The solid and dashed curves represent the simulation results[17] of the PSD and the asymptotic PSD of codes based on the MSRDS$^{(3)}$ criterion respectively; the dashed-dot curves represent spectra of codes based on the bimode criterion. This figure demonstrates that both the scrambling polynomial and the quotient selection criterion affect spectral performance, and that use of the MSRDS$^{(3)}$ criterion, which is more complex to implement, results in superior spectrum compression (up to about 9dB improvement at low frequencies) when compared to the bimode criterion which is easier to implement.
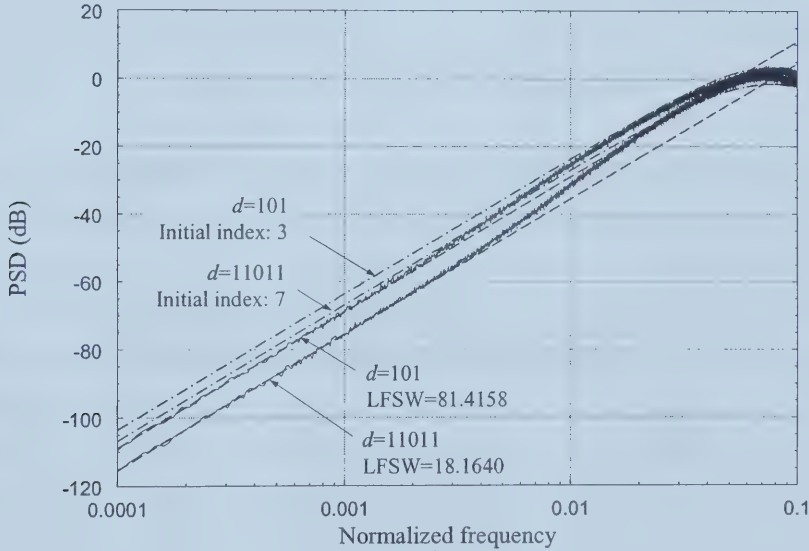
Figure 5.5 Power Spectra of GS 4/8 dc^2-free multimode codes generated
through the MSRDS$^{(3)}$ criterion (solid and dashed curves) and
the bimode technique (dash-dotted curves).

Figures 5.6 and 5.7 depict the comparison of simulated power spectra (solid curves) and asymptotic power spectra (dashed curves) of GS 14/20 dc^2-free multimode codes generated through MSW$^{(2)}$ and MSRDS$^{(3)}$ criteria, and GS 9/20 dc^3-free multimode codes generated through MSW$^{(3)}$ and MSRDS$^{(4)}$ criteria, respectively. From these figures, it can be seen that the simulation results agree with theoretical expectations. The simulated PSDs and asymptotic

---

[17] Simulated PSDs and LFSWs in this thesis are based on over 5×10^6 channel symbols.

PSDs of multimode codes using the $MSW^{(K)}$ criteria $(K = 2, 3)$ presented in Figures 5.6 and 5.7 are very close to the corresponding exact PSDs and asymptotic PDSs of the same multimode codes presented in Figures 5.2 and 5.3. Also these figures demonstrate that in this application the $MSRDS^{(3)}$ and $MSRDS^{(4)}$ criteria result in about 9.5 dB and 14 dB PSD performance improvement at the normalized frequency 10^{-4}, compared to the $MSW^{(2)}$ and $MSW^{(3)}$ criterion respectively. Furthermore, to compare PSD performance of GS dc^K-free multimode codes using the $MSRDS^{(K)}$ criterion and the corresponding monomode codes with the same codeword length and the same order of spectral null K, the asymptotic PSDs of the best monomode codes which are evaluated in Section 3.3 are also included in Figures 5.6 and 5.7. These figures illustrate that given K and codeword length 20, GS dc^K-free multimode codes using the $MSRDS^{(K+1)}$ criterion realize both higher code rate (10% and 20% increase in code rate for dc^2-free and dc^3-free codes respectively) and better PSD performance (about 3 dB and 2.5 dB decrease of PSDs at the normalized frequency 10^{-4} for dc^2-free and dc^3-free codes respectively) than the best dc^K-free monomode codes. Note that dc^K-free monomode codes are dominant in the literature.

Further examples are given in Figures 5.8 and 5.9 where dc^2-free monomode and multimode codes and dc^3-free monomode and multimode codes with codeword length 24 are compared. Similar to Figures 5.6 and 5.7, these figures indicate that, for a given word length, GS multimode codes can not only realize higher rate than the corresponding monomode codes with the same order spectral null at dc, but can also ensure much larger suppression of low-frequency components than the best spectral-null monomode codes of the same order. In these examples, there is about 5.3 dB and 10 dB increased PSD suppression at the normalized frequency 10^{-4} for the dc^2- and dc^3-free codes respectively. Also Figures 5.8 and 5.9 demonstrate that the new $MSRDS^{(K+1)}$ selection criterion is superior to the $MSW^{(K)}$ criterion. With these codes, there is about 12.3 dB and 16.6 dB increased PSD suppression at the normalized frequency 10^{-4} for the dc^2- and dc^3-free codes respectively.

Note that the $MSRDS^{(K+1)}$ criterion is not limited to dc^K-free codes when $K > 1$; it can also be applied to dc^1-free codes for generation of a deeper spectral null. In Figure 5.10 the $MSRDS^{(2)}$ selection criterion applied to a dc^1-free multimode code is compared with the previously published $MSW^{(1)}$ selection criterion. The simulated PSDs and asymptotic PSDs of GS 21/24 dc^1-free multimode codes show that the $MSRDS^{(2)}$ criterion results in about 2.7 dB improvement in suppression of low- frequency components in this application.

Figure 5.6 Comparison of simulated PSDs of GS 14/20 dc^2-free multimode codes (d=101011) and asymptotic PSD of the 12/20 dc^2-free monomode code with the lowest LFSW.

Figure 5.7 Comparison of simulated PSDs of GS 9/20 dc^3-free multimode codes (d=11000101) and asymptotic PSD of the 5/20 dc^3-free monomode code with the lowest LFSW.

Figure 5.8 Comparison of simulated PSDs of GS 17/24 dc^2-free multimode codes (d=100111) and asymptotic PSD of the 15/24 dc^2-free monomode code with the lowest LFSW.
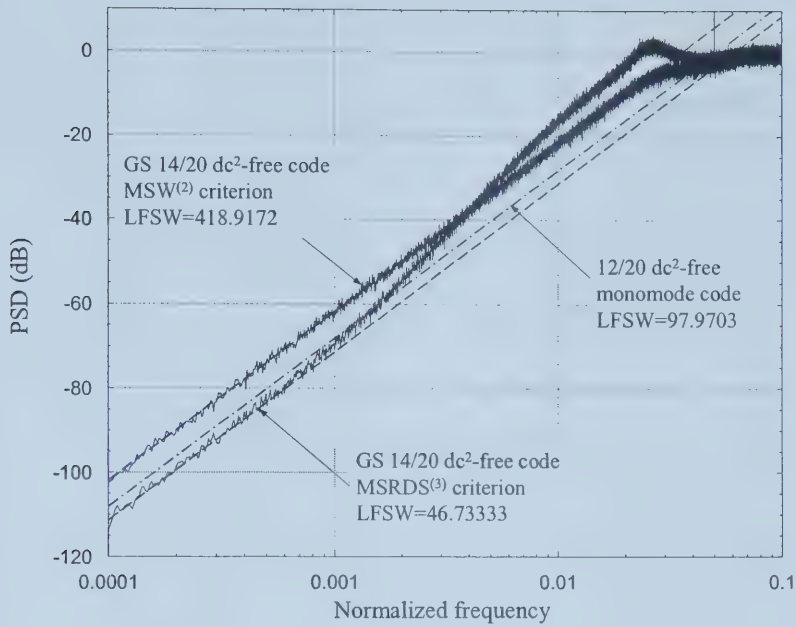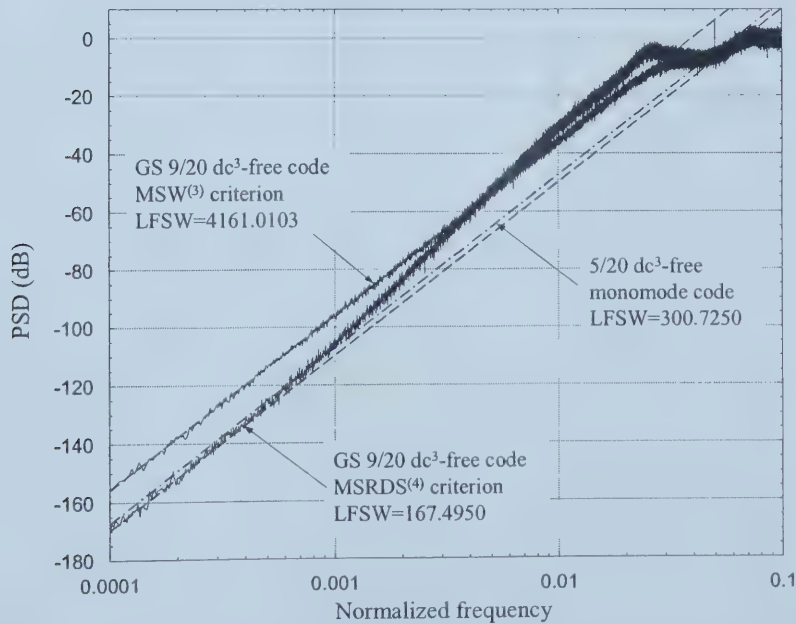

Figure 5.9 Comparison of simulated PSDs of GS 12/24 dc^3-free multimode codes (d=100010101) and asymptotic PSD of the 9/24 dc^3-free monomode code with the lowest LFSW.
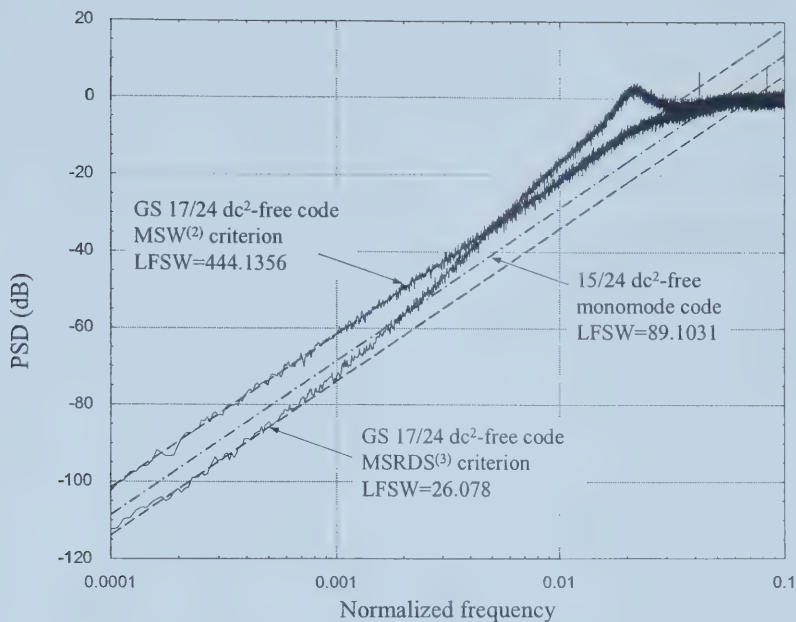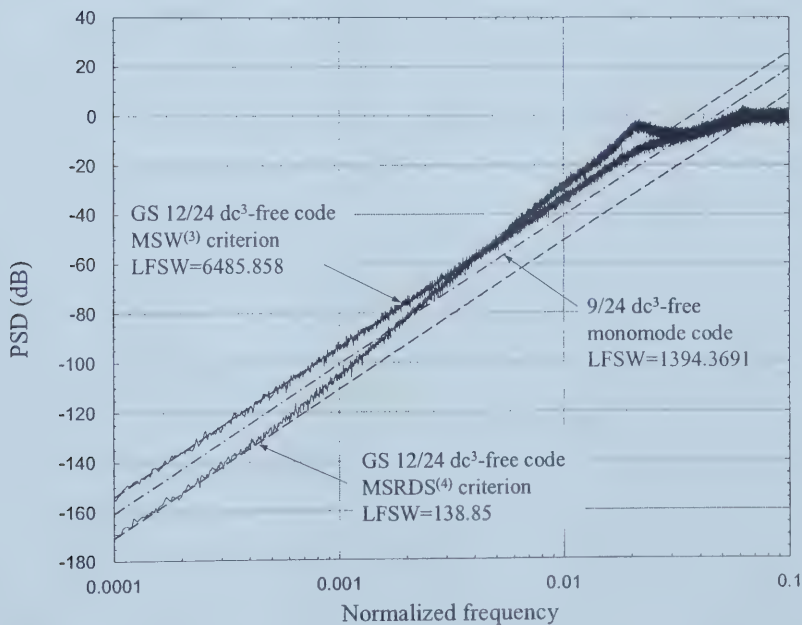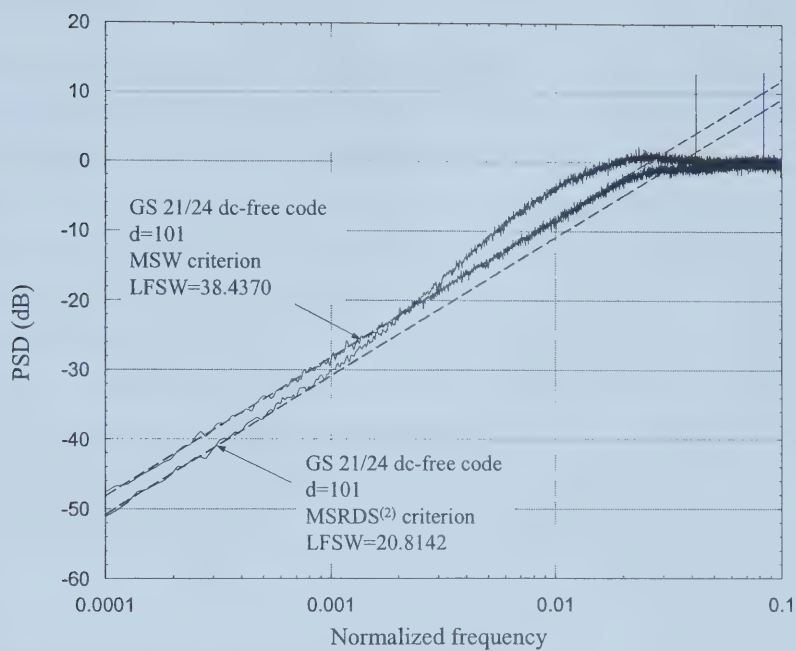
Figure 5.10 Comparison of simulated PSDs of GS 21/24 dc[1]-free multimode codes (d=101) using the MSW[1] and MSRDS[2] criteria.

Chapter 6
CONCLUSION

This thesis has investigated several characteristics of high-order spectral-null codes, proposed a new performance metric for evaluation of an arbitrary high-order spectral-null code, developed construction methods for generation of multimode codes, and introduced a new class of high-order spectral null codes, high-order spectral-null multimode codes. Foundation for the work in this thesis includes: definition of constrained coding and high-order spectral-null coding principles introduced in Chapter 1; definition of high-order spectral-null codes and the model of their encoders described in Chapter 2; characteristics of high-order spectral-null codes and power spectrum calculation of fixed- and variable-length codes reviewed in Chapters 2 and 3 respectively; and description of multimode codes, the guided scrambling multimode coding technique, and characteristics in GS presented in Chapter 4. In the remainder of this chapter the research contributions and results of this thesis are summarized and possible further work is suggested.

6.1 Summary of Research Contributions

Although characteristics of high-order spectral-null codes have been extensively developed in the literature, there exist many open problems regarding these codes. In Chapter 2, an improved lower bound was derived for minimum word length of high-order zero-disparity codes given the order of spectral null, and an improved lower bound for redundancy of high-order zero-disparity codes given word length and the order of spectral null was aslo derived. A new method was introduced for enumerating codewords with specified values of WD and WDS. Enumeration rules were given, and two new efficient algorithms for evaluating the cardinality of codewords based on this enumeration method were derived. Calculation results demonstrate that the number of visited nodes of the new Algorithm I is about an order of magnitude lower than a recently published algorithm. For the cases when the word length and desired WD are fixed and the desired WDS falls in a large range of values, the new Algorithm II is significantly more efficient than Algorithm I. The new algorithms can find application in enumerating codewords for dc^2-free codes, particularly when the codeword length is large and when sets of codewords with a range of values of WDS are of interest. In addition, a fast algorithm was introduced for enumerating the cardinality of dc^K-free words given the word length. Enumeration results regarding the number of available codewords were tabulated.

133

The cutoff frequency is widely used for measuring the width of spectral nulls of dc^1-free codes. However, in Chapter 3 it was demonstrated that this metric cannot be generalized to be a valid measure of spectrum rejection for dc^K-free codes $(K > 1)$. In its place, the low-frequency spectrum-weight was introduced as a general metric for comparing the performance of codes with an arbitrarily high-order spectral-null at dc. In Chapter 3, it was shown that the asymptotic low-frequency spectral components of HOSN codes are approximately linear versus frequency on a logarithmic scale, and that such components can be closely approximated with knowledge of the order of the spectral-null and the LFSW of the code. Furthermore, the LFSW was shown to be equivalent to the zero-frequency spectral content of the sequence of the same order RDS values. The result for symbol-by-symbol coding was extended to block coding, and the characterization of the sequence of Kth-order RDS values of dc^K-free codes was illustrated.

Also in Chapter 3, a closed-form expression for the LFSW of Kth-order zero-disparity codes $(K \geq 1)$ was presented. This LFSW is determined by the order of spectral null K, the number of required codewords, the codeword length, and the sum of the squared values of the Kth-order moments of the codewords. A convenient enumeration approach for calculation of LFSW of high-order zero-disparity codes was outlined, and the average, minimal, and maximal asymptotic low-frequency spectral components of such codes were investigated. Simple expressions were given for the LFSW of first-order zero-disparity codes, low-disparity dc^1-free codes, and maxentropic dc^1-free codes. It was shown that the product of redundancy and square root of LFSW of maxentropic codes is approximately constant when the absolute bound of $RDS^{(1)}$ values is large.

Throughout Chapter 3, examples were given to show the validity and the reliability of this new performance metric. Low-frequency spectrum-weight is a general and effective metric for determining the low-frequency spectrum compression of high-order spectral-null codes. It can also be employed for designing a high-order spectral-null code with large rejection of low-frequency components.

Chapter 5 demonstrated that the guided scrambling line coding technique can be used to generate high-order spectral-null multimode codes. A state-dependent encoding method for block coded sequences was proposed in order to construct rate-efficient high-order spectral-null codes that can be implemented in a simple manner. To construct a dc^K-free code, it is sufficient that each source word is mapped into a codeword selection set that includes at least one pair of complementary dc^{K-1}-free words. In guided scrambling, scrambling polynomials play a key role in determining the mapping between source words and codewords. In Chapter 4, it was shown

that the necessary and sufficient conditions for the generation of complementary quotients in each GS selection set are that the GS base scrambling polynomials have degree less than or equal to the number of augmenting bits A and have even weight. There are 2^{A-1} base polynomials that satisfy these conditions. The relationship between augmenting bit patterns and complementary quotients and corresponding remainders was given. Appropriate base polynomials suitable for dc^K-free codes were obtained through further computer search, and some of these polynomials were listed in Appendix 3. In Chapter 5, the resulting code rate was tabulated for word lengths up to 32 and the order of spectral null up to 4.

In Chapter 5 it was shown that a state-dependent fixed-length encoder can be modeled with an equivalent state-independent variable-length encoder. The low-frequency spectrum-weight of the encoded sequences generated through these equivalent encoders has been derived. The analysis of this low-frequency spectrum-weight provides insight into the construction of good high-order spectral-null codes as well as good dc^1-free codes with a large rejection of spectral components at low frequencies. Selection criteria were discussed, and it was shown that a serial selection mechanism can be effective in practice when the GS code with large quotient selection sets. Spectral results of various GS dc^K-free multimode codes generated through the $MSRDS^{(K+1)}$ selection criterion demonstrate that GS coding can be used to yield superior spectral performance and higher rate codes than other high-order spectral-null encoding techniques developed to date.

6.2 Suggested Further Work

This thesis has developed systematic methods to construct high-order spectral-null multimode codes and to evaluate the power spectrum performance of high-order spectral-null codes. There remains potential further work regarding extensive analysis of high-order spectral-null codes, some areas of investigation of which are outlined below.

A. It would be advantageous if methods could be developed to more easily determine base scrambling polynomials suitable for high-order spectral-null multimode GS codes. It would also be useful to evaluate the relationship between base polynomials and spectral performance of high-order spectral-null multimode codes so that appropriate polynomials could be chosen to meet the spectrum requirements.

B. In GS dc^K-free encoding discussed in this thesis, due to use of dc^{K-1}-free words with $RDS^{(r)} = 0$, $r = 1, \cdots, K-1$, bounding of only one parameter, $RDS^{(K)}$, is considered. Design of

GS encoders that consider more than one parameter may result in other enhancement, as indicated below:

- development of more efficient high-order spectral-null multimode codes. The dc^K-free multimode codes proposed in this thesis require that each selection set has a pair of complementary dc^{K-1}-free words, and the selection criteria proposed ensure that $RDS^{(K)}$ values in the coded sequences are bounded. More rate-efficient dc^K-free multimode codes $(K \geq 2)$ could be implemented by considering bounding multiple orders of RDS.

- development of codes that generate spectral nulls at different frequencies. It has been shown that bounding $RDS^{(K)}$ at a specific frequency results in a Kth-order spectral null at this frequency [25].

- development of dc-free runlength-limited codes. In this instance, efforts must be made to ensure the constraints in both time-domain and frequency-domain are met.

C. Further analysis of the low-frequency spectrum-weight performance metric is possible, including:

- development of an analytic method to evaluate LFSW of dc^1-free bimode and multimode codes;

- evaluation of LFSW of maxentropic dc^K-free codes $(K \geq 2)$;

- development of a performance metric that combines LFSW and code rate for dc^K-free codes $(K \geq 1)$.

D. It should be possible to develop search criteria that will lead to larger suppression of low-frequency components. Selection criteria developed in this thesis for GS codes are quite straightforward because they select codewords on a word-by-word basis. More complicated selection criteria that look ahead over several upcoming source words [94] may result in better spectrum performance [77].

E. Development of GS spectral null codes with embedded error control properties.

- It has been shown that matched spectral-null codes [24] provide a distance gain when applied in partial response channels where the frequencies at which spectral nulls of coded sequences exist is coincident with the frequencies at which the channel transfer function is zero. The error correction ability that results from this distance gain for high-order spectral-null multimode codes needs to be evaluated.

- Combining spectral null codes with error-correction codes is another interesting topic that shows promise.

BIBLIOGRAPHY

[1]   K. A. S. Immink, *Codes for Mass Data Storage Systems*. Amsterdam, The Netherlands: Shannon Foundation Publishers, 1999.

[2]   E. R. Berlekamp, "The technology of error-correcting codes," *Proceedings of the IEEE*, vol. 68, pp. 564-593, May 1980.

[3]   Special issue on signal processing for high-density storage channels, *IEEE Journal on. Selected Areas in Communications*, vol. 19, no. 4, Apr. 2001.

[4]   K. A. S. Immink, P. H. Siegel, and J. K. Wolf, "Codes for digital recorders," *IEEE Transactions on Information Theory*, vol. 44, no. 6, pp. 2260-2299, Oct. 1998.

[5]   S. B. Wicker and V. K. Bhargava, Eds, *Reed-Solomon Codes and Their Applications*, IEEE Press, 1994.

[6]   C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: turbo codes," *Proceedings of IEEE International Conference on Communications*, Geneva, Switherland, 1993, pp. 1064-1070.

[7]   D. J. C. MacKay, "Near Shannon limit performance of low density parity check codes," *Electronics Letters*, vol. 33, no. 6, pp. 457-458, Mar. 1997.

[8]   B. H. Marcus, P. H. Siegel and J. K. Wolf, "Finite-state modulation codes for data storage," *IEEE Journal on. Selected Areas in Communications*, vol. 10, no. 1, pp. 5-37, Jan. 1992.

[9]   K. A. S. Immink, "Runlength-limited sequences," *Proceedings of the IEEE*, vol. 78, no. 11, pp. 1745-1759, Nov. 1990.

[10]   K. A. S. Immink, "A survey of codes for optical disk recording," *IEEE Journal on. Selected Areas in Communications*, vol. 19, no.4, pp. 756-764, Apr. 2001.

[11]   R. D. Cideciyan, F. Dolivo, R. Hermann, W. Hirt, and W. Schott, "A PRML system for digital magnetic recoding," *IEEE Journal on. Selected Areas in Communications*, vol. 10, no. 1, pp. 38-56, Jan. 1992.

[12]   P. Kabal and S. Pasupathy, "Partial-response signaling," *IEEE Transactions on Communications*, vol. COM-23, pp. 921-934, no. 9, Sept. 1975.

[13]   G. D. Forney, Jr., "Maximum likelihood sequence detection in the presence of intersymbol interference," *IEEE Transactions on Information Theory*, vol. IT-18, no. 3, pp. 363-378, May 1972.

[14] H. K. Thapar and A. M. Patel, "A class of partial response systems for increasing storage density in magnetic recoding," *IEEE Transactions on Magnetics*, vol. MAG-23, no. 5, pp. 3666-3668, Sept. 1987.

[15] G. D. Forney, Jr., "The Viterbi algorithm," *Proceedings of the IEEE*, vol. 61, pp. 268-278, Mar. 1973.

[16] K. W. Cattermole, "Principles of digital line coding," *International Journal on Electronics*, vol. 55, no. 1, pp. 3-33, 1983.

[17] K. A. S. Immink, "The digital versatile disc (DVD): system requirements and channel coding," *Journal of Society of Motion Picture and Television Engineers (SMPTE)*, vol. 105, pp. 483-489, Aug. 1996.

[18] K. A. S. Immink, "EFM coding: squeezing the last bits," *IEEE Transactions on Consumer Electronics*, vol. 43, no. 3, pp. 491-495, Aug. 1997.

[19] R. M. Roth, "On runlength-limited coding with dc control," *IEEE Transactions on Communications*, vol. 48, no. 3, pp. 351-358, Mar. 2000.

[20] V. Braun and K. A. S. Immink, "An enumerative coding technique for dc-free runlength-limited sequences," *IEEE Transactions on Communications*, vol. 48, no. 12, pp. 2024-2031, Dec. 2000.

[21] K. A. S. Immink and L. Patrovics, "Performance assessment of dc-free multimode codes," *IEEE Transactions on Communications*, vol. 45, no. 3, pp. 293-299, Mar. 1997.

[22] K. A. S. Immink, "Spectrum shaping with binary DC^2-constrained channel codes," *Phillips Journal of Research*, vol. 40, pp. 40-53, 1985.

[23] K. A. S. Immink and G. Beenker, "Binary transmission codes with high order spectral zeros at zero frequency," *IEEE Transactions on Information Theory*, vol. IT-33, no. 3, pp. 452-454, May 1987.

[24] R. Karabed and P. H. Siegel, "Matched spectral-null codes for partial-response channels," *IEEE Transactions on Information Theory*, vol. 37, no.3, pp. 818-855, May 1991.

[25] E. Eleftheriou and R. D. Cideciyan, "On codes satisfying Mth-order running digital sum constraints," *IEEE Transactions on Information Theory*, vol. 37, no. 5, pp. 1294-1313, Sept. 1991.

[26] C. M. Monti and G. L. Pierobon, "Codes with a multiple spectral null at zero frequency," *IEEE Transactions on Information Theory*, vol. 35, no. 2, pp. 463-472, Mar. 1989.

[27]   R. M. Roth, P. H. Siegel and A. Vardy, "High-order spectral-null codes - constructions and bounds," *IEEE Transactions on Information Theory*, vol. 40, no. 6, pp. 1826-1840, Nov. 1994.

[28]   V. Skachek, T. Etzion, and R. M. Roth, "Efficient encoding algorithm for third-order spectral-null codes," *IEEE Transactions on Information Theory*, vol. 44, no.2, pp. 846-851, Mar. 1998.

[29]   G. Freiman and S. Litsyn, "Asymptotically exact bounds on the size of high-order spectral-null codes," *IEEE Transactions on Information Theory*, vol. 45, no. 6, pp. 1798-1807, Sept. 1999.

[30]   L. G. Tallini and B. Bose, "On efficient high-order spectral-null codes," *IEEE Transactions on Information Theory*, vol. 45, no. 7, pp. 2594-2601, Nov. 1999.

[31]   B. Vasic, G. Djordjevic, and M. Tosic, "Loose composite constraint codes and their application in DVD," *IEEE Journal on. Selected Areas in Communications*, vol. 19, no. 4, pp. 765-773, Apr. 2001.

[32]   C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, pp. 379-423, July 1948.

[33]   T. M. Chien, "Upper bound on the efficiency of dc-constrained codes," *Bell System Technical Journal*, vol. 49, pp. 2267-2287, Nov. 1970.

[34]   G. L. Cariolaro, G. L. Pierobon, and G. P. Tronca, "Analysis of codes and spectra calculations," *International Journal on Electronics*, vol. 55, pp. 35-79, 1983.

[35]   A. Papoulis, *Probability, Random Variables, and Stochastic Processes*. New York: McGraw-Hill, ch. 9.4, 1965.

[36]   W. R. Bennett, "Statistics of regenerative digital transmission," *Bell System Technical Journal*, vol. 37, pp. 1501-1521, Nov. 1958.

[37]   L. E. Franks, *Signal Theory*. Prentice-Hall, Inc., Englewood Cliffs, New Jersey, ch.8, 1969.

[38]   W. A. Gardener, "Characterization of cyclostationary random signal processes," *IEEE Transactions on Information Theory*, vol. IT-21, no. 1, pp. 4-14, Jan. 1975.

[39]   J. G. Proakis and D. G. Manolakis, *Digital Signal Processing – Principle, Algorithms, and Applications*. 3rd ed., Upper Saddle River, New Jersey: Prentice-Hall, 1996.

[40]   E. A. Lee and D. G. Messerschmitt, *Digital Comminication*. Boston, MA: Kluwer Academic Publishers, ch.10, 1988.

[41] R. M. Brooks and A. Jessop, "Line coding for optical fiber systems," *International Journal on Electronics*, vol. 55, pp. 81-121, 1983.

[42] D. B. Waters, "Line codes for metallic cable systems," *International Journal on Electronics*, vol. 55, pp. 159-169, 1983.

[43] G. L. Pierobon, "Codes for zero spectral density at zero frequency," *IEEE Transactions on Information Theory*, vol. IT-30, no. 2, pp. 435-439, Mar. 1984.

[44] P. A. Franaszek, "Sequence-state coding for digital transmission," *Bell System Technical Journal*, vol. 47, pp. 143-157, Jan. 1968.

[45] Z. Kohavi, *Switching and Finite Automata Theory*, New York: McGraw-Hill, p. 281, 1970.

[46] G. Bilardi, R. Padovani, and G. L. Pierobon, "Spectral analysis of Markov chains with applications," *IEEE Transactions on Communications*, vol. COM-31, no. 7, pp. 853-861, July 1983.

[47] R. Adler, D. Coppersmith, and M. Hassner, "Algorithms for sliding block codes," *IEEE Transactions on Information Theory*, vol. IT-29, no. 1, pp. 5-22, Jan. 1983.

[48] G. H. Hardy and E. M. Wright, *An Introduction to the Theory of Numbers*. Oxford: Oxford University, ch.21, 1979.

[49] L. K. Hua, *Introduction to Number Theory*. Berlin: Springer-Verlag, ch.18, 1982.

[50] P. Borwein and C. Ingalls, "The Prouhet-Tarry-Escott problem revisited," *L'Enseignement Mathématique*, vol. 40, pp. 3-27, 1994.

[51] L. M. G. M. Tolhuizen, K. A. S. Immink, and H. D. Hollmann, "Constructions and properties of block codes for partial-response channels," *IEEE Transactions on Information Theory*, vol. 41, no. 6, pp. 2019-2026, Nov. 1995.

[52] D. W. Boyd, "On a problem of Byrnes concerning polynomials with restricted coefficients," *Mathematics of Computation*, vol. 66, pp. 1697-1704, 1997.

[53] H. L. Dorwart and O. E. Brown, "The Tarry-Escott problem," *American Mathematics Monthly*, vol. 44, pp. 613-625, 1937.

[54] J. S. Byrnes, "Problem on polynomials with restricted coefficients arising from questions in antenna array theory," *Recent Advances in Fourier Analysis and Its Applications*, J. S. Byrnes and J. F. Byrnes, Eds. Norwell, MA: Kluwer, pp. 677-678, 1990.

[55] D. W. Boyd, "On a problem of Byrnes concerning polynomials with restricted coefficients, II," *Mathematics of Computation*, Electronic Version, pp. 1-13, May 11, 2001.

[56] E. R. Berlakamp, *Algebraic Coding Theory*. New York: McGraw-Hill, ch.4, 1968.

[57] D. E. Knuth, *The Art of Computer Programming, vol. 1*. Reading, MA: Addison-Wesley Publishing Company, 1968.

[58] I. S. Gradshteyn and I. M. Ryzgik, translated by Yu. V. Geronimus and M. Yu. Tseytlin, *Table of Integrals, Series, and Products*. 4th ed., Academic Press Inc., 1963.

[59] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*. Amsterdam: North-Holland, ch.17, 1977.

[60] J. Riordan, *An Introduction to Combinatorial Analysis*. Princeton, NJ: Princeton University Press, 1980.

[61] K. A. S. Immink, "Performance of simple binary dc-constrained codes," *Phillips Journal of Research*, vol. 40, pp. 1-21, 1985.

[62] J. Justesen, "Information rates and power spectra of digital codes," *IEEE Transactions on Information Theory*, vol. IT-28, no. 3, pp. 457-472, May 1982.

[63] K. W. Cattermole and J. J. O'Reilly, *Problems of Randomness in Communication Engineering, vol. 2*. London: Pentech Press, 1984.

[64] H. Yasuda, "Calculation of autocorrelation function and entropy of pulse sequences by means of transition probability matrices," *Electronics and Communications in Japan*, vol. 54-A, pp. 17-23, Nov. 1971.

[65] B. S. Bosik, "The spectral density of a coded digital signal," *Bell System Technical Journal*, vol. 51, pp. 921-932, Apr. 1972.

[66] G. L. Cariolaro and G. P. Tronca, "Spectra of block coded digital signals," *IEEE Transactions on Communications*, vol. COM-22, no. 10, pp. 1555-1564, Oct. 1974.

[67] A. V. Oppenheim and A. S. Willsky, *Signals and Systems*. Upper Saddle River, NJ: Prentice-Hall, 1997.

[68] S. M. Ross, *Introduction to Probability Models*. 5th Ed., San Diego, CA: Academic Press, 1993.

[69] J. N. Franklin and J. R. Pierce, "Spectra and efficiency of binary codes without dc," *IEEE Transactions on Communications*, vol. COM-20, no. 12, pp. 1182-1184, Dec. 1972.

[70] G. L. Cariolaro and G. L. Pierobon, "Stationary symbol sequences from variable-length word sequences," *IEEE Transactions on Information Theory*, vol. IT-23, no. 2, pp. 243-253, Mar. 1977.

[71] G. L. Cariolaro, G. L. Pierobon, and S. G. Pupolin, "Spectral analysis of variable-length coded digital signals," *IEEE Transactions on Information Theory*, vol. IT-28, no. 3, pp. 473-481, May 1982.

[72] V. I. Johannes, "Comments on 'compatible high-density bipolar codes: an unrestricted transmission plan for PCM carriers'," *IEEE Transactions on Communications*, vol. COM-20, no. 1, pp. 78-79, Feb. 1972.

[73] V. A. Dieuliis and F. P. Preparata, "Spectrum shaping with alphabetic codes with finite autocorrelation sequence," *IEEE Transactions on Communications*, vol. COM-26, no. 4, pp. 474-478, Apr. 1978.

[74] G. D. Forney and A. R. Calderbank, "Coset codes for partial response channels; or coset codes with spectral nulls," *IEEE Transactions on Information Theory*, vol. 35, no. 5, pp. 925-943, Sept. 1989.

[75] V. Braun and A. J. E. M. Janssen, "On the low-frequency suppression performance of dc-free runlength-limited modulation codes," *IEEE Transactions on Consumer Electronics*, vol. 42, no. 4, pp. 939-945, Nov. 1996.

[76] M. C. Chiu, "DC-free error-correcting codes based on convolutional codes," *IEEE Transactions on Communications*, vol. 49, no. 4, pp. 609-619, Apr. 2001.

[77] K. A. S. Immink, "EFMPlus: The coding format of the multimedia compact disc," *IEEE Transactions on Consumer Electronics*, vol. 41, no. 3, pp. 491-497, Aug. 1995.

[78] J. W. Harris and H. Stocker, *Handbook of Mathematics and Computational Science*. New York: Springer-Verlag, 1998.

[79] I. J. Fair, W. D. Grover, W. A. Krzymien, and R. I. MacDonald, "Guided scrambling: a new line coding technique for high bit rate fiber optic transmission systems," *IEEE Transactions on Communications*, vol. 39, no. 2, pp. 289-297, Feb. 1991.

[80] K. J. Kerpez, "The power spectral density of maximum entropy charge constrained sequences," *IEEE Transactions on Information Theory*, vol. 35, no.3, pp. 692-695, May 1989.

[81] L. G. Tallini, R. M. Capocelli, and B. Bose, "Design of new efficient balanced codes," *IEEE Transactions on Information Theory*, vol. 42, no. 3, pp. 790-802, May 1996.

[82] K. A. S. Immink, "Construction of binary dc-constrained codes," *Phillips Journal of Research*, vol. 40, pp. 22-39, 1985.

[83] D. E. Knuth, "Efficient balanced codes," *IEEE Transactions on Information Theory*, vol. IT-32, no. 1, pp.51-53, Jan. 1986.

[84] S. Al-Bassam and B. Bose, "On balanced codes," *IEEE Transactions on Information Theory*, vol. 36, no. 2, pp. 406-408, Mar. 1990.

[85] F. K. Bowers, U.S. Patent 2 957 947, 1960.

[86] R. O. Carter, "Low-disparity binary coding system," *Electronics Letters*, vol. 1, no. 3, pp. 67-68, May 1965.

[87] R. H. Deng and M. A. Herro, "DC-free coset codes," *IEEE Transactions on Information Theory*, vol. 34, no. 4, pp. 786-792, July 1988.

[88] A. Kunisa, S. Takahasi, and N. Itoh, "Digital modulation method for recordable digital video disc," *IEEE Transactions on Consumer Electronics*, vol. 42, no. 3, pp. 820-825, Aug. 1996.

[89] I. J. Fair, Q. Wang, and V. K. Bhargava, "Polynomials for guided scrambling line codes," *IEEE Journal on Selected Areas in Communications*, vol. 13, no. 3, pp. 499-509, Apr. 1995.

[90] I. J. Fair, Q. Wang, and V. K. Bhargava, "Characteristics of guided scrambling encoders and their coded sequences," *IEEE Transactions on Information Theory*, vol. 43, no. 1, pp. 342-347, Jan. 1997.

[91] I. J. Fair, V. K. Bhargava, and Q. Wang, "Evaluation of the power spectral density of guided scrambling coded sequences," *IEE Proceedings-Communications*, vol. 144, no. 2, Apr. 1997.

[92] J. B. Fraleigh, *A First Course in Abstract Algebra*. Readings, MA: Addison-Wesley Publishing, 4th Ed., 1989.

[93] R. Lidl and H. Niederreiter, *Finite Fields*. Cambridge, United Kingdom: Cambridge University Press, p. 27, 1997.

[94] G. V. Jacoby, "A new look-ahead code for increasing data density," *IEEE Transactions on Magnetics*, vol. MAG-13, no. 5, pp. 1202-1204, Sept. 1977.

Appendix 1
Proof of Theorem 3.2

For a given $H_{\lambda\mu}(z)$, it is convenient to evaluate $H_x(z)$ based on (3.46). As described in Section 3.1.3, for a mutually independent length sequence $\{L_m\}$, the sequence $y(h\Lambda)$, $h = 0, 1, \cdots$, converges to $y(\infty\Lambda) = \Lambda/L$. According to (3.41), there is:

$$H_{\lambda\mu}(z) = \sum_{h=-\infty}^{\infty} \left[R_{\lambda\mu}(h\Lambda T_b) - R_{\lambda\mu}(\infty\Lambda T_b) \right] z^{-h\Lambda}$$

$$= R_{\lambda\mu}(0) - R_{\lambda\mu}(\infty\Lambda T_b) + \sum_{h=1}^{\infty} \left[R_{\lambda\mu}(h\Lambda T_b) - R_{\lambda\mu}(\infty\Lambda T_b) \right] z^{-h\Lambda}$$

$$+ \sum_{h=-\infty}^{-1} \left[R_{\lambda\mu}(h\Lambda T_b) - R_{\lambda\mu}(\infty\Lambda T_b) \right] z^{-h\Lambda}. \tag{A1-1}$$

Also from (3.39), (3.43), and (3.44), it can be found that

$$\sum_{h=1}^{\infty} \left[R_{\lambda\mu}(h\Lambda T_b) - R_{\lambda\mu}(\infty\Lambda T_b) \right] z^{-h\Lambda}$$

$$= \sum_{h=1}^{\infty} \left[y(h\Lambda - \lambda) m_\lambda' m_\mu - y(\infty\Lambda) m_\lambda' m_\mu \right] z^{-h\Lambda}$$

$$= \sum_{h=1}^{\infty} x(h - \lambda') z^{-h\Lambda} m_\lambda' m_\mu \tag{A1-2}$$

where $x(h) = y(h\Lambda) - y(\infty\Lambda)$, which is an absolute summable sequence, and $\lambda' = \lambda/\Lambda$. Define $X(z) = \sum_{h=0}^{\infty} x(h) z^{-h}$ to be the z-transform of $x(h)$.

- When $\lambda' = 1$

$$\sum_{h=1}^{\infty} \left[R_{\lambda\mu}(h\Lambda T_b) - R_{\lambda\mu}(\infty\Lambda T_b) \right] z^{-h\Lambda}$$

$$= \left[\sum_{h-1=0}^{\infty} x(h-1) z^{-(h-1)\Lambda} \right] z^{-\Lambda} m_\lambda' m_\mu$$

$$= X(z^\Lambda) z^{-\Lambda} m_\lambda' m_\mu ; \tag{A1-3}$$

- When $\lambda' \geq 2$

$$\sum_{h=1}^{\infty} \left[R_{\lambda\mu}(h\Lambda T_b) - R_{\lambda\mu}(\infty\Lambda T_b) \right] z^{-h\Lambda}$$

$$= \left[\sum_{h=\lambda'}^{\infty} x(h-\lambda') z^{-(h-\lambda')\Lambda} + \sum_{h=1}^{\lambda'-1} x(h-\lambda') z^{-(h-\lambda')\Lambda} \right] z^{-\lambda'\Lambda} m_\lambda' m_\mu$$

$$= \left[X(z^\Lambda) + \sum_{h=1}^{\lambda'-1} -y(\infty\Lambda) z^{-(h-\lambda')\Lambda} \right] z^{-\lambda'\Lambda} m_\lambda' m_\mu$$

$$= \left[X(z^\Lambda)z^{-\lambda} - (\Lambda/L)\sum_{h=1}^{\lambda'-1} z^{-h\Lambda} \right] m'_\lambda m_\mu. \tag{A1-4}$$

Similarly,

$$\sum_{h=-\infty}^{-1} \left[\boldsymbol{R}_{\lambda\mu}(h\Lambda T_b) - \boldsymbol{R}_{\lambda\mu}(\infty\Lambda T_b) \right] z^{-h\Lambda}$$

$$= \sum_{h=1}^{\infty} \left[\boldsymbol{R}_{\lambda\mu}(-h\Lambda T_b) - \boldsymbol{R}_{\lambda\mu}(\infty\Lambda T_b) \right] z^{h\Lambda}$$

$$= \sum_{h=1}^{\infty} [y(h\Lambda - \mu) - y(\infty\Lambda)] z^{h\Lambda} m'_\lambda m_\mu$$

$$= \sum_{h=1}^{\infty} x(h - \mu')z^{-h(-\Lambda)} m'_\lambda m_\mu \tag{A1-5}$$

where $\mu' = \mu/\Lambda$.

- When $\mu' = 1$

$$\sum_{h=-\infty}^{-1} \left[\boldsymbol{R}_{\lambda\mu}(h\Lambda T_b) - \boldsymbol{R}_{\lambda\mu}(\infty\Lambda T_b) \right] z^{-h\Lambda}$$

$$= \left[\sum_{h-1=0}^{\infty} x(h-1)z^{-(h-1)(-\Lambda)} \right] z^\Lambda m'_\lambda m_\mu$$

$$= X(z^{-\Lambda})z^\Lambda m'_\lambda m_\mu ; \tag{A1-6}$$

- When $\mu' \geq 2$

$$\sum_{h=-\infty}^{-1} \left[\boldsymbol{R}_{\lambda\mu}(h\Lambda T_b) - \boldsymbol{R}_{\lambda\mu}(\infty\Lambda T_b) \right] z^{-h\Lambda}$$

$$= \left[\sum_{h=\mu'}^{\infty} x(h-\mu')z^{-(h-\mu')(-\Lambda)} + \sum_{h=1}^{\mu'-1} x(h-\mu')z^{-(h-\mu')(-\Lambda)} \right] z^{\mu'\Lambda} m'_\lambda m_\mu$$

$$= \left[X(z^{-\Lambda}) + \sum_{h=1}^{\mu'-1} -y(\infty\Lambda)z^{-(h-\mu')(-\Lambda)} \right] z^{-\mu'\Lambda} m'_\lambda m_\mu$$

$$= \left[X(z^{-\Lambda})z^\mu - (\Lambda/L)\sum_{h=1}^{\mu'-1} z^{h\Lambda} \right] m'_\lambda m_\mu. \tag{A1-7}$$

By replacing (A1-3), (A1-4), (A1-6), and (A1-7) into (A1-1), expression (3.45) follows when $\lambda', \mu' \geq 1$ where $X(z)$ for general cases will be evaluated below. Note that $X(z)$ has been evaluated in [70] only for the case of $\Lambda = 1$, and that the expression for $X(z)$ given there contains typographical errors.

From the definition of cumulative length probabilities $y(k)$ in (3.39) and from the independence of word lengths in $\{L_m\}$, a differential equation exists:

$$y(k) = \sum_{\lambda} p_{\lambda} y(k-\lambda), \qquad k > 0. \tag{A1-8}$$

Due to $k = h\Lambda$ ($h = 0, 1, \cdots$), it can be shown that $y(0) = 1$, $y(h\Lambda) = 0$ when $h < 0$, and $y(\infty\Lambda) = \Lambda/L$ (Λ is the g.c.d of word length, $\Lambda \geq 1$, and L is the average word length). Since $x(h) = y(h\Lambda) - y(\infty\Lambda)$, for $h > 0$, from (A1-8) another differential equation follows:

$$x(h) = \sum_{\lambda} p_{\lambda} y(h\Lambda - \lambda) - y(\infty\Lambda)$$

$$= \sum_{\lambda} p_{\lambda} \left[y(h\Lambda - \lambda) - y(\infty\Lambda) \right]$$

$$= \sum_{\lambda} p_{\lambda} x(h - \lambda').$$

Also there are $x(0) = 1 - \Lambda/L$ and $x(h) = -\Lambda/L$ when $h < 0$. So

$$X(z) = \sum_{h=0}^{\infty} x(h) z^{-h} = x(0) + \sum_{h=1}^{\infty} x(h) z^{-h}$$

$$= 1 - \Lambda/L + \sum_{h=1}^{\infty} \sum_{\lambda} p_{\lambda} x(h - \lambda') z^{-h}$$

$$= 1 - \Lambda/L + \sum_{\lambda} p_{\lambda} \sum_{h=1}^{\infty} x(h - \lambda') z^{-h}$$

$$= 1 - \Lambda/L + p_{\lambda'=1} \sum_{h-1=0}^{\infty} x(h-1) z^{-(h-1)} z^{-1} + \sum_{\lambda' \geq 2} p_{\lambda} \sum_{h=1}^{\lambda'-1} x(h - \lambda') z^{-h}$$

$$+ \sum_{\lambda' \geq 2} p_{\lambda} \sum_{h-\lambda'=0}^{\infty} x(h - \lambda') z^{-(h-\lambda')} z^{-\lambda'}$$

$$= 1 - \Lambda/L + p_{\lambda'=1} X(z) z^{-1} - (\Lambda/L) \sum_{\lambda' \geq 2} p_{\lambda} \sum_{h=1}^{\lambda'-1} z^{-h} + \sum_{\lambda' \geq 2} p_{\lambda} X(z) z^{-\lambda'}.$$

By means of $\sum_{\lambda} p_{\lambda} = 1$, it can be obtained that:

$$X(z) = 1 + X(z) \sum_{\lambda} p_{\lambda} z^{-\lambda'} - (\Lambda/L) \left[\sum_{\lambda} p_{\lambda} + \sum_{\lambda' \geq 2} p_{\lambda} \sum_{h=1}^{\lambda'-1} z^{-h} \right]$$

$$= 1 + X(z) \sum_{\lambda} p_{\lambda} z^{-\lambda'} - (\Lambda/L) \left[p_{\lambda'=1} + \sum_{\lambda' \geq 2} p_{\lambda} \sum_{h=0}^{\lambda'-1} z^{-h} \right]$$

$$= 1 + X(z) \sum_{\lambda} p_{\lambda} z^{-\lambda'} - (\Lambda/L) \sum_{\lambda} p_{\lambda} \sum_{h=0}^{\lambda'-1} z^{-h}.$$

Furthermore,

$$X(z)\left(1-\sum_{\lambda}p_{\lambda}z^{-\lambda'}\right)=1-(\Lambda/L)\sum_{\lambda}p_{\lambda}\sum_{h=0}^{\lambda'-1}z^{-h} \qquad (A1\text{-}9)$$

in which

$$1-\sum_{\lambda}p_{\lambda}z^{-\lambda'}=\sum_{\lambda}p_{\lambda}(1-z^{-\lambda'})$$

$$=(1-z^{-1})\sum_{\lambda}p_{\lambda}(1+z^{-1}+\cdots+z^{-(\lambda'-1)}) \qquad (A1\text{-}10)$$

and

$$1-(\Lambda/L)\sum_{\lambda}p_{\lambda}\sum_{h=0}^{\lambda'-1}z^{-h}=(\Lambda/L)\sum_{\lambda}p_{\lambda}\lambda'-(\Lambda/L)\sum_{\lambda}p_{\lambda}\sum_{h=0}^{\lambda'-1}z^{-h}$$

$$=(\Lambda/L)\sum_{\lambda}p_{\lambda}\left(\lambda'-\sum_{h=0}^{\lambda'-1}z^{-h}\right)$$

$$=(\Lambda/L)\left[p_{\lambda'=1}(1-z^{0})+\sum_{\lambda'\geq 2}p_{\lambda}\left(\sum_{h=0}^{\lambda'-1}(1-z^{-h})\right)\right]$$

$$=(\Lambda/L)\left[\sum_{\lambda'\geq 2}p_{\lambda}\sum_{h=1}^{\lambda'-1}(1-z^{-1})(1+z^{-1}+\cdots+z^{-(h-1)})\right]$$

$$=(1-z^{-1})(\Lambda/L)\left[\sum_{\lambda'\geq 2}p_{\lambda}\sum_{h=1}^{\lambda'-1}(1+z^{-1}+\cdots+z^{-(h-1)})\right]$$

$$=(1-z^{-1})(\Lambda/L)\sum_{\lambda'\geq 2}p_{\lambda}\left[(\lambda'-1)\cdot 1+(\lambda'-2)\cdot z^{-1}+\cdots+z^{-(\lambda'-2)}\right]. \qquad (A1\text{-}11)$$

- When $z\neq 1$, replacing (A1-10) and (A1-11) into (A1-9) yields:

$$X(z)=\frac{(\Lambda/L)\sum_{\lambda'\geq 2}p_{\lambda}\left[(\lambda'-1)\cdot 1+(\lambda'-2)\cdot z^{-1}+\cdots+z^{-(\lambda'-2)}\right]}{\sum_{\lambda}p_{\lambda}(1+z^{-1}+\cdots+z^{-(\lambda'-1)})}; \qquad (A1\text{-}12)$$

- When $z=1$, evaluation of $X(1)$ is obtained by considering the limit of $X(z)$:

$$X(1)=\lim_{z\to 1}X(z)$$

$$=\lim_{z\to 1}\frac{1-(\Lambda/L)\sum_{\lambda}p_{\lambda}\sum_{h=0}^{\lambda'-1}z^{-h}}{1-\sum_{\lambda}p_{\lambda}z^{-\lambda'}}$$

$$=\frac{(\Lambda/L)\sum_{\lambda'\geq 2}p_{\lambda}(1/2)\lambda'(\lambda'-1)}{\sum_{\lambda}p_{\lambda}\lambda'}. \qquad (A1\text{-}13)$$

Expressions (A1-12) and (A1-13) complete the proof. $\square$

Appendix 2
Proof of Theorem 3.7

The exact LFSW value of maxentropic dc^1-free codes is given by (3.87) in Section 3.3.4. Let $H_{\text{RDS}^{(1)}}(0) = \mathcal{H}_1 - \mathcal{H}_2$, where

$$\mathcal{H}_1 = \frac{4}{(B+1)^2} \sum_{j=1}^{B} \mu_j \sum_{i=1}^{B} \mu_i \sum_{k=1}^{B} \frac{2\sin(i2k\gamma)\sin(j2k\gamma)}{1 - \frac{\cos(2k\gamma)}{\cos(\gamma)}} \tag{A2-1}$$

and

$$\mathcal{H}_2 = \frac{2}{B+1} \sum_{j=1}^{B} (B+1-j) \sum_{i=1}^{B} \mu_i \sin(i\gamma), \tag{A2-2}$$

and let

$$r'_{i,j} = \sum_{k=1}^{B} \frac{2\sin(i2k\gamma)\sin(j2k\gamma)}{1 - \frac{\cos(2k\gamma)}{\cos(\gamma)}}. \tag{A2-3}$$

A simple expression of $r'_{i,j}$ has not yet been found, but the following conjecture is given.

Conjecture: $r'_{i,j}$ is approximately represented as:

$$r'_{i,j} \approx \begin{cases} \xi \sin(i\gamma)\cos(j\gamma), & \text{if } i \leq j \\ \xi \cos(i\gamma)\sin(j\gamma), & \text{otherwise} \end{cases}, \tag{A2-4}$$

where ξ is independent of the values of i, j, k, but is dependent on B.

This conjecture is based on the observation below.

Observation:

Let $\mathbf{R}'_B$ be the $B \times B$ matrix with entry $r'_{i,j}$ in row i and column j for the given value of B.

(a) B = 2

$$\mathbf{R}'_2 = \left[r'_{i,j} \right]_{2\times2} = \begin{bmatrix} 4.500000 & 2.598176 \\ 2.598176 & 4.500000 \end{bmatrix}$$

$$\left[\frac{r'_{i,j}}{\sin(i\gamma)\cos(j\gamma)} \right]_{2\times2} = \begin{bmatrix} 10.392304 & 10.392304 \\ 3.464101 & 10.392304 \end{bmatrix}$$

$$\left[\frac{r'_{i,j}}{\cos(j\gamma)\sin(i\gamma)} \right]_{2\times2} = \begin{bmatrix} 10.392304 & 3.464101 \\ 10.392304 & 10.392304 \end{bmatrix}$$

(b) B = 3

$$\mathbf{R}'_3 = \left[r'_{i,j} \right]_{3\times3} = \begin{bmatrix} 6.828427 & 5.226252 & 2.828427 \\ 5.226252 & 9.656854 & 5.226251 \\ 2.828427 & 5.226251 & 6.828426 \end{bmatrix}$$

$$\left[\frac{r'_{i,j}}{\sin(i\gamma)\cos(j\gamma)}\right]_{3\times3} = \begin{bmatrix} 19.313707 & 19.313707 & 19.313708 \\ 7.999999 & 19.313707 & 19.313707 \\ 3.313708 & 7.999999 & 19.313708 \end{bmatrix}$$

$$\left[\frac{r'_{i,j}}{\cos(j\gamma)\sin(i\gamma)}\right]_{3\times3} = \begin{bmatrix} 19.313707 & 7.999999 & 3.313708 \\ 19.313708 & 19.313707 & 7.999999 \\ 19.313708 & 19.313708 & 19.313708 \end{bmatrix}$$

(c) B = 4

$$R'_4 = \left[r'_{i,j}\right]_{4\times4} = \begin{bmatrix} 9.045085 & 7.694208 & 5.590170 & 2.938926 \\ 7.694208 & 14.635254 & 10.633134 & 5.590169 \\ 5.590170 & 10.633134 & 14.635254 & 7.694208 \\ 2.938926 & 5.590169 & 7.694208 & 9.045084 \end{bmatrix}$$

$$\left[\frac{r'_{i,j}}{\sin(i\gamma)\cos(j\gamma)}\right]_{4\times4} = \begin{bmatrix} 30.776834 & 30.776834 & 30.776834 & 30.776833 \\ 13.763818 & 30.776834 & 30.776833 & 30.776834 \\ 7.265425 & 16.245983 & 30.776834 & 30.776833 \\ 3.249196 & 7.265424 & 13.763817 & 30.776835 \end{bmatrix}$$

$$\left[\frac{r'_{i,j}}{\cos(i\gamma)\sin(j\gamma)}\right]_{4\times4} = \begin{bmatrix} 30.776834 & 13.763818 & 7.265425 & 3.249196 \\ 30.776834 & 30.776834 & 16.245983 & 7.265424 \\ 30.776834 & 30.776833 & 30.776834 & 13.763817 \\ 30.776833 & 30.776834 & 30.776833 & 30.776835 \end{bmatrix}.$$

To determine the value of ξ, it is convenient to let $i = j = 1$. From (A2-3) there is:

$$r'_{1,1} = \cos(\gamma)\sum_{k=1}^{B}\frac{2\sin^2(2k\gamma)}{\cos(\gamma)-\cos(2k\gamma)}$$

$$= 2\cos(\gamma)\sum_{k=1}^{B}\frac{\cos^2(\gamma)+\sin^2(\gamma)-\cos^2(2k\gamma)}{\cos(\gamma)-\cos(2k\gamma)}$$

$$= 2\cos(\gamma)\sum_{k=1}^{B}\frac{[\cos(\gamma)-\cos(2k\gamma)][\cos(\gamma)+\cos(2k\gamma)]+\sin^2(\gamma)}{\cos(\gamma)-\cos(2k\gamma)}$$

$$= 2B\cos^2(\gamma)+2\cos(\gamma)\sum_{k=1}^{B}\cos(2k\gamma)+2\cos(\gamma)\sin^2(\gamma)\sum_{k=1}^{B}\frac{1}{\cos(\gamma)-\cos(2k\gamma)}$$

Because $\gamma = \dfrac{\pi}{2(B+1)}$, $\displaystyle\sum_{k=1}^{B}\cos(2k\gamma) = 0$. Therefore:

$$r'_{1,1} = 2B\cos^2(\gamma)+2\cos(\gamma)\sin^2(\gamma)\sum_{k=1}^{B}\frac{1}{\cos(\gamma)-\cos(2k\gamma)}.$$

When $B \gg 1$, due to Euler's summation formula [57, p.108], there exists:

$$\sum_{k=1}^{B}\frac{1}{\cos(\gamma)-\cos(2k\gamma)} \approx \int_{1}^{B+1}\frac{dk}{\cos(\gamma)-\cos(2k\gamma)} .$$

Lemma A-1 [58, p. 148]:

$$\int\frac{dx}{a+b\cos x} = \frac{1}{\sqrt{b^2-a^2}}\ln\frac{\sqrt{b^2-a^2}\,\mathrm{tg}\frac{x}{2}+a+b}{\sqrt{b^2-a^2}\,\mathrm{tg}\frac{x}{2}-a-b} , \quad a^2 < b^2 . \tag{A2-5}$$

By means of Lemma A-1, it is straightforward to show that:

$$\int_{1}^{B+1}\frac{dk}{\cos(\gamma)-\cos(2k\gamma)} = \frac{1}{2\gamma\sin(\gamma)}\ln\frac{1-\cos(\gamma)}{\cos(\gamma)-\cos(2\gamma)} .$$

It can also be shown that:

$$\sum_{k=1}^{B}\frac{1}{\cos(\gamma)-\cos(2k\gamma)} \approx \frac{\ln 3}{2\gamma\sin(\gamma)} , \quad B \gg 1 .$$

Finally it can be obtained that:

$$r'_{1,1} \approx 2B\cos^2(\gamma) + 2\cos(\gamma)\sin^2(\gamma)\frac{\ln 3}{2\gamma\sin(\gamma)}$$

$$\approx 2B\cos^2(\gamma) = \sin(1\cdot\gamma)\cos(1\cdot\gamma)\cdot 2B\frac{\cos(\gamma)}{\sin(\gamma)} , \quad B \gg 1 .$$

From (A2-4), there is:

$$\xi \approx \frac{r'_{1,1}}{\sin(1\cdot\gamma)\cos(1\cdot\gamma)} \approx 2B\frac{\cos(\gamma)}{\sin(\gamma)} .$$

Therefore:

$$r'_{i,j} \approx \begin{cases} 2B\dfrac{\cos\gamma}{\sin\gamma}\sin(i\gamma)\cos(j\gamma), & \text{if } i \leq j \\[4mm] 2B\dfrac{\cos\gamma}{\sin\gamma}\cos(i\gamma)\sin(j\gamma), & \text{otherwise} . \end{cases} \tag{A2-6}$$

By replacing (A2-3) and (A2-6) into (A2-1), it is found that:

$$\mathcal{H}_1 = \frac{4}{(B+1)^2}\frac{B\cos(\gamma)}{\sin(\gamma)}\left[\underbrace{\sum_{j=1}^{B}(B+1-j)\sin(j\gamma)\cos(j\gamma)\sum_{i=1}^{j}(B+1-i)2\sin^2(i\gamma)}_{h_{1,1}}\right.$$

$$\left. + \underbrace{\sum_{j=1}^{B}(B+1-j)\sin^2(j\gamma)\sum_{i=j+1}^{B}(B+1-i)2\sin(i\gamma)\cos(i\gamma)}_{h_{1,2}}\right] . \tag{A2-7}$$

There are numerous formulas regarding the calculation of summations and integrals in [58], some of which are useful for the following derivations. It can be shown that:

$$\sum_{i=1}^{j}(B+1-i)2\sin^2(i\gamma)$$

$$=(B+1)j-\frac{1}{2}j(j+1)-(B+1)\frac{\cos(j+1)\gamma\sin j\gamma}{\sin\gamma}+\frac{(j+1)\sin(2j+1)\gamma}{2\sin\gamma}-\frac{1-\cos(j+1)2\gamma}{4\sin^2\gamma} \quad \text{(A2-8)}$$

$$\sum_{i=j+1}^{B}(B+1-i)2\sin(i\gamma)\cos(i\gamma)$$

$$=\frac{(B+1)\cos(\gamma)}{2\sin(\gamma)}-\frac{(B+1)\sin((j+1)\gamma)\sin(j\gamma)}{\sin(\gamma)}+\frac{\sin((j+1)2\gamma)}{4\sin^2(\gamma)}-\frac{(j+1)\cos((2j+1)\gamma)}{2\sin(\gamma)}.$$

$$\text{(A2-9)}$$

By combining (A2-8) and (A2-9) with (A2-7), there is:

$$h_{1,1}=\frac{B+1}{2}\left[\underbrace{\sum_{j=1}^{B}(B+1)j\sin(2j\gamma)}_{h_{1,1,1}}-\underbrace{\frac{1}{2}\sum_{j=1}^{B}j(j+1)\sin(2j\gamma)}_{h_{1,1,2}}-\underbrace{\frac{B+1}{\sin\gamma}\sum_{j=1}^{B}\cos((j+1)\gamma)\sin(j\gamma)\sin(2j\gamma)}_{h_{1,1,3}}\right.$$

$$+\underbrace{\frac{1}{2\sin\gamma}\sum_{j=1}^{B}(j+1)\sin((2j+1)\gamma)\sin(2j\gamma)}_{h_{1,1,4}}-\underbrace{\frac{1}{4\sin^2\gamma}\sum_{j=1}^{B}\sin(2j\gamma)}_{h_{1,1,5}}+\underbrace{\frac{1}{4\sin^2\gamma}\sum_{j=1}^{B}\cos((j+1)2\gamma)\sin(2j\gamma)}_{h_{1,1,6}}$$

$$-\underbrace{\frac{B+1}{2}\sum_{j=1}^{B}j^2\sin(2j\gamma)}_{h_{1,1,7}}+\underbrace{\frac{1}{4}\sum_{j=1}^{B}j^2(j+1)\sin(2j\gamma)}_{h_{1,1,8}}+\underbrace{\frac{B+1}{2\sin\gamma}\sum_{j=1}^{B}j\cos((j+1)\gamma)\sin(j\gamma)\sin(2j\gamma)}_{h_{1,1,9}}$$

$$\left.-\underbrace{\frac{1}{4\sin\gamma}\sum_{j=1}^{B}j(j+1)\sin((2j+1)\gamma)\sin(2j\gamma)}_{h_{1,1,10}}+\underbrace{\frac{1}{8\sin^2\gamma}\sum_{j=1}^{B}j\sin(2j\gamma)}_{h_{1,1,11}}-\underbrace{\frac{1}{8\sin^2\gamma}\sum_{j=1}^{B}j\cos((j+1)2\gamma)\sin(2j\gamma)}_{h_{1,1,12}}\right]$$

and:

$$H_{1,2}=\underbrace{\sum_{j=1}^{B}(B+1-j)\sin^2(j\gamma)\frac{(B+1)\cos(\gamma)}{2\sin(\gamma)}}_{h_{1,2,1}}-\underbrace{\sum_{j=1}^{B}(B+1-j)\sin^2(j\gamma)\frac{(B+1)\sin((j+1)\gamma)\sin(j\gamma)}{\sin(\gamma)}}_{h_{1,2,2}}$$

$$+\underbrace{\sum_{j=1}^{B}(B+1-j)\sin^2(j\gamma)\frac{\sin((j+1)2\gamma)}{4\sin^2(\gamma)}}_{h_{1,2,3}}-\underbrace{\sum_{j=1}^{B}(B+1-j)\sin^2(j\gamma)\frac{(j+1)\cos((2j+1)\gamma)}{2\sin(\gamma)}}_{h_{1,2,4}}.$$

In the following derivations, assume that $B\gg1$ such that $\sin(\gamma)\approx\gamma$ and $\cos(\gamma)\approx1$. Then:

- $h_{1,1,1} = \dfrac{(B+1)^2 \cos(\gamma)}{\sin(\gamma)} \approx \dfrac{(B+1)^2}{\gamma}$;

- $h_{1,1,2} = \dfrac{1}{2}\displaystyle\sum_{j=1}^{B} j^2 \sin(j2\gamma) + \dfrac{1}{2}\sum_{j=1}^{B} j\sin(j2\gamma)$

  $\approx \dfrac{1}{2}\displaystyle\int_{1}^{B+1} j^2 \sin(j2\gamma)dj + \dfrac{(B+1)\cos(\gamma)}{4\sin(\gamma)}$

  $\approx \dfrac{(B+1)^2}{4\gamma} - \dfrac{1}{4\gamma^3} + O(B^2)$;

- $h_{1,1,3} = \dfrac{B+1}{2\sin(\gamma)}\displaystyle\sum_{j=1}^{B}\big[\sin((2j+1)\gamma) - \sin(\gamma)\big]\sin(2j\gamma)$

  $= \dfrac{B+1}{4\sin(\gamma)}\displaystyle\sum_{j=1}^{B}\big[\cos(\gamma) - \cos((4j+1)\gamma) - \cos((2j-1)\gamma) + \cos((2j+1)\gamma)\big]$

  $\approx \dfrac{B(B+1)\cos(\gamma)}{4\sin(\gamma)} - \dfrac{B+1}{4\sin(\gamma)}\displaystyle\int_{1}^{B+1}\cos((4j+1)\gamma)dj - \dfrac{B+1}{4\sin(\gamma)}\int_{1}^{B+1}\cos((2j-1)\gamma)dj$

  $\quad + \dfrac{B+1}{4\sin(\gamma)}\displaystyle\int_{1}^{B+1}\cos((2j+1)\gamma)dj$

  $\approx \dfrac{(B+1)^2}{4\gamma} + O(B^2)$;

- $h_{1,1,4} = \dfrac{1}{4\sin(\gamma)}\displaystyle\sum_{j=1}^{B}(j+1)(\cos(\gamma) - \cos((4j+1)\gamma))$

  $\approx \dfrac{B(B+1)\cos(\gamma)}{8\sin(\gamma)} + \dfrac{B\cos(\gamma)}{4\sin(\gamma)} - \dfrac{1}{4\sin(\gamma)}\displaystyle\int_{1}^{B+1} j\cos((4j+1)\gamma)dj - \dfrac{1}{4\sin(\gamma)}\int_{1}^{B+1}\cos((4j+1)\gamma)dj$

  $\approx \dfrac{(B+1)^2}{8\gamma} + O(B^2)$;

- $h_{1,1,5} = \dfrac{\cos\gamma}{4\sin^3\gamma} \approx \dfrac{1}{4\gamma^3}$;

- $h_{1,1,6} = \dfrac{1}{8\sin^2(\gamma)}\displaystyle\sum_{j=1}^{B}\sin((4j+2)\gamma) - \dfrac{1}{8\sin^2(\gamma)}\sum_{j=1}^{B}\sin(2\gamma)$

  $\approx \dfrac{1}{8\sin^2(\gamma)}\displaystyle\int_{1}^{B+1}\sin((4j+2)\gamma)dj - \dfrac{B\cos(\gamma)}{4\sin(\gamma)}$

  $= O(B^2)$;

- $h_{1,1,7} \approx \dfrac{B+1}{2}\displaystyle\int_{1}^{B+1} j^2 \sin(2j\gamma)dj$

$$\approx \frac{(B+1)^3}{4\gamma} - \frac{B+1}{4\gamma^3} + O(B^2) \,;$$

- $h_{1,1,8} = \dfrac{1}{4}\displaystyle\sum_{j=1}^{B} j^3 \sin(2j\gamma) + \dfrac{1}{4}\displaystyle\sum_{j=1}^{B} j^2 \sin(2j\gamma)$

$$\approx \frac{1}{4}\int_{.1}^{B+1} j^3 \sin(2j\gamma)dj + \frac{1}{4}\int_{.1}^{B+1} j^2 \sin(2j\gamma)dj$$

$$\approx \frac{(B+1)^3}{8\gamma} - \frac{3(B+1)}{16\gamma^3} + O(B^3)$$

- $h_{1,1,9} = \dfrac{B+1}{4\sin(\gamma)}\displaystyle\sum_{j=1}^{B} j\big[\sin((2j+1)\gamma) - \sin(\gamma)\big]\sin(2j\gamma)$

$$= \frac{B+1}{4\sin(\gamma)}\sum_{j=1}^{B} j\sin((2j+1)\gamma)\sin(2j\gamma) - \frac{B+1}{4}\sum_{j=1}^{B} j\sin(2j\gamma)$$

$$\approx \frac{B(B+1)^2 \cos(\gamma)}{16\sin(\gamma)} - \frac{B+1}{8\sin(\gamma)}\int_{.1}^{B+1} j\cos((4j+1)\gamma)dj - \frac{(B+1)^2 \cos(\gamma)}{8\sin(\gamma)}$$

$$\approx \frac{(B+1)^3}{16\gamma} + O(B^3) \,;$$

- $h_{1,1,10} = \dfrac{1}{8\sin(\gamma)}\displaystyle\sum_{j=1}^{B} j^2\big[\cos(\gamma) - \cos((4j+1)\gamma)\big] + \dfrac{1}{8\sin(\gamma)}\displaystyle\sum_{j=1}^{B} j\big[\cos(\gamma) - \cos((4j+1)\gamma)\big]$

$$\approx \frac{B(B+1)(2B+1)\cos(\gamma)}{48\sin(\gamma)} - \frac{1}{8\sin(\gamma)}\int_{.1}^{B+1} j^2 \cos((4j+1)\gamma)dj + \frac{B(B+1)\cos(\gamma)}{16\sin(\gamma)}$$

$$- \frac{1}{8\sin(\gamma)}\int_{.1}^{B+1} j\cos((4j+1)\gamma)dj$$

$$\approx \frac{(B+1)^3}{24\gamma} - \frac{B+1}{64\gamma^3} + O(B^3) \,;$$

- $h_{1,1,11} = \dfrac{1}{8\sin^2\gamma}\cdot\dfrac{(B+1)\cos(\gamma)}{2\sin(\gamma)}$

$$\approx \frac{B+1}{16\gamma^3} + O(B^3) \,;$$

- $h_{1,1,12} = \dfrac{1}{16\sin^2(\gamma)}\displaystyle\sum_{j=1}^{B} j\big[\sin((4j+2)\gamma) - \sin(2\gamma)\big]$

$$\approx \frac{1}{16\sin^2(\gamma)}\int_{.1}^{B+1} j\sin((4j+2)\gamma)dj - \frac{B(B+1)\cos(\gamma)}{16\sin(\gamma)}$$

$$\approx -\frac{B+1}{64\gamma^3} + O(B^3) \,.$$

Finally, it can be found that:

$$h_{1,1} = \frac{B+1}{2}(h_{1,1,1} - h_{1,1,2} - h_{1,1,3} + h_{1,1,4} - h_{1,1,5} + h_{1,1,6})$$

$$- (h_{1,1,7} - h_{1,1,8} - h_{1,1,9} + h_{1,1,10} - h_{1,1,11} + h_{1,1,12})$$

$$\approx \frac{5(B+1)}{32\gamma^3} - \frac{(B+1)^3}{24\gamma} + O(B^3). \tag{A2-10}$$

Considering the value of $H_{1,2}$ there is:

- $$h_{1,2,1} = \frac{(B+1)\cos(\gamma)}{4\sin(\gamma)} \sum_{j=1}^{B}(B+1-j)(1-\cos(2j\gamma))$$

$$= \frac{B(B+1)^2 \cos(\gamma)}{4\sin(\gamma)} - \frac{B(B+1)^2 \cos(\gamma)}{8\sin(\gamma)} - \frac{(B+1)^2 \cos(\gamma)}{4\sin(\gamma)} \sum_{j=1}^{B}\cos(2j\gamma)$$

$$+ \frac{(B+1)\cos(\gamma)}{4\sin(\gamma)} \sum_{j=1}^{B} j\cos(2j\gamma)$$

$$\approx \frac{(B+1)^3}{8\gamma} - \frac{B+1}{8\gamma^3} + O(B^3).$$

- $$h_{1,2,2} = \frac{B+1}{4\sin(\gamma)} \sum_{j=1}^{B}(B+1-j)(1-\cos(2j\gamma))\left[\cos(\gamma) - \cos((2j+1)\gamma)\right]$$

$$= \frac{B(B+1)^2 \cos(\gamma)}{4\sin(\gamma)} - \frac{B(B+1)^2 \cos(\gamma)}{8\sin(\gamma)} + \frac{(B+1)\cos(\gamma)}{4\sin(\gamma)} \sum_{j=1}^{B} j\cos(2j\gamma)$$

$$- \frac{(B+1)^2 \cos(\gamma)}{4\sin(\gamma)} \sum_{j=1}^{B}\cos((2j+1)\gamma) + \frac{B+1}{4\sin\gamma} \sum_{j=1}^{B} j\cos((2j+1)\gamma)$$

$$+ \frac{(B+1)^2}{4\sin(\gamma)} \sum_{j=1}^{B}\cos 2j\gamma \cos((2j+1)\gamma) - \frac{B+1}{4\sin(\gamma)} \sum_{j=1}^{B} j\cos(2j\gamma)\cos((2j+1)\gamma)$$

$$\approx \frac{B(B+1)^2 \cos(\gamma)}{8\sin(\gamma)} + \frac{(B+1)\cos(\gamma)}{4\sin(\gamma)}(\frac{B+1}{2} - \frac{1}{2\sin^2(\gamma)})$$

$$- \frac{(B+1)^2}{4\sin(\gamma)} \int^{B+1}\cos((2j+1)\gamma)dj + \frac{B+1}{4\sin(\gamma)} \int^{B+1} j\cos((2j+1)\gamma)dj$$

$$+ \frac{(B+1)^2}{4\sin(\gamma)} \int^{B+1}\cos(2j\gamma)\cos((2j+1)\gamma)dj - \frac{B+1}{4\sin(\gamma)} \int^{B+1} j\cos(2j\gamma)\cos((2j+1)\gamma)dj$$

$$\approx \frac{3(B+1)^3}{16\gamma} - \frac{B+1}{4\gamma^3} + O(B^3);$$

- $$h_{1,2,3} = \frac{1}{8\sin^2(\gamma)} \sum_{j=1}^{B}(B+1-j)(1-\cos(2j\gamma))\sin((j+1)2\gamma)$$

154

$$= \frac{1}{8\sin^2(\gamma)} \sum_{j=1}^{B} (B+1)\sin((j+1)2\gamma) - \frac{1}{8\sin^2(\gamma)} \sum_{j=1}^{B} j\sin((j+1)2\gamma)$$

$$- \frac{1}{16\sin^2(\gamma)} \sum_{j=1}^{B} (B+1-j)\big[\sin((4j+2)\gamma) + \sin(2\gamma)\big]$$

$$\approx \frac{3(B+1)}{64\gamma^3} + O(B^3);$$

- $$h_{1,2,4} = \frac{1}{4\sin(\gamma)} \sum_{j=1}^{B} (B+1-j)(1-\cos(2j\gamma))(j+1)\cos((2j+1)\gamma)$$

$$= \frac{B+1}{4\sin(\gamma)} \sum_{j=1}^{B} (j+1)\cos((2j+1)\gamma) - \frac{B+1}{4\sin(\gamma)} \sum_{j=1}^{B} (j+1)\cos(2j\gamma)\cos((2j+1)\gamma)$$

$$- \frac{1}{4\sin(\gamma)} \sum_{j=1}^{B} j(j+1)\cos((2j+1)\gamma) + \frac{1}{4\sin(\gamma)} \sum_{j=1}^{B} j(j+1)\cos(2j\gamma)\cos((2j+1)\gamma)$$

$$\approx \frac{B+1}{64\gamma^3} - \frac{(B+1)^3}{48\gamma} + O(B^3).$$

Therefore:

$$h_{1,2} = h_{1,2,1} - h_{1,2,2} + h_{1,2,3} - h_{1,2,4}$$

$$\approx \frac{5(B+1)}{32\gamma^3} - \frac{(B+1)^3}{24\gamma} + O(B^3). \tag{A2-11}$$

From (A2-10) and (A2-11), it is straightforward to show that

$$\mathcal{H}_1 = \frac{4}{(B+1)^2} \frac{B\cos(\gamma)}{\sin(\gamma)} (h_{1,1} + h_{1,2})$$

$$\approx \frac{60 - 4\pi^2}{3\pi^4} (B+1)^4 + O(B^3). \tag{A2-12}$$

Also, from (A2-2), there exists:

$$\mathcal{H}_2 = \frac{2}{B+1} \sum_{j=1}^{B} (B+1-j) \sum_{i=1}^{B} \mu_i \sin(i\gamma)$$

$$= B \sum_{i=1}^{B} (B+1-i)\sin^2 i\gamma$$

$$= \frac{B}{2} \sum_{i=1}^{B} (B+1-i)(1-\cos 2i\gamma)$$

$$= O(B^3). \tag{A2-13}$$

Based on the values of $\mathcal{H}_1$ and $\mathcal{H}_2$ in (A2-12) and (A2-13), Theorem 3.7 follows. $\qquad\square$

155

Appendix 3

List of Base Scrambling Polynomials for GS Dc^K-Free Multimode Codes ($2 \leq K \leq 4$, $N \leq 32$)

For GS M/N dc^K-free multimode codes ($K \geq 2$), the base scrambling polynomials listed below ensure that every selection set contains at least one pair of complementary dc^{K-1}-free words. From Theorem 4.1, the highest degree D of a base polynomial of a GS M/N dc^K-free multimode code is $N - M$. For convenience, available polynomials are represented by their coefficients; the polynomial $d(x) = x^D + d_{D-1}x^{D-1} + \cdots + d_2 x^2 + d_1 x + 1$ is represented by $1\, d_{D-1} \cdots d_2\, d_1\, 1$. Due to limit in searching time and space limitations of this appendix, in some cases, only a partial list of base scrambling polynomials is given. These cases are noted by an upper limit on D.

GS 1/4 dc²-free codes

D=1
11

D=2
101

D=3
1001

GS 2/6 dc²-free codes

D=1
11

D=2
101

D=3
1001
1111

D=4
10001
10111
11011
11101

GS 4/8 dc²-free codes

D=2
101

D=4
11011

GS 5/10 dc²-free codes

D=2
101

D=3
1001

D=4
10111
11011
11101

D=5
100111
101011
101101
110011
110101
111001

GS 7/12 dc²-free codes

D=3
1001

D=4
10111
11101

D=5
100111
101011
110011
110101
111001

GS 8/14 dc²-free codes

D=2
101

D=3
1001

D=4
10001
10111
11101

D=5
100111
101011
101101
110011
110101
111001

D=6
1000111
1001011
1001101
1010011
1011001
1011111
1100011
1100101
1101001
1101111
1110001
1110111
1111011
1111101

GS 10/16 dc^2-free codes

D=4
10001

D=5
100111
101101
111001

D=6
1000111
1001011
1001101
1010011
1011001
1011111
1100011
1100101
1101001
1101111
1110001
1110111
1111011
1111101

GS 12/18 dc^2-free codes

D=5
100111
101011
101101
110101
111001

D=6
1000111
1001011
1001101
1010011
1011001
1011111
1100101
1101001
1101111
1110001
1110111
1111011
1111101

GS 14/20 dc^2-free codes

D=5
101011
110101

D=6
1010011
1100101

GS 15/22 dc^2-free codes

D=5
100111

101011
101101
110011
110101
111001

D=6
1000111
1001011
1001101
1010011
1011001
1011111
1100101
1101001
1101111
1110001
1111011
1111101

D=7
10000111
10001011
10001101
10010011
10010101
10011001
10100011
10101001
10101111
10110001
10110111
10111011
11000101
11001001
11010001
11010111
11011101
11100001
11100111
11101011
11101101
11110101

GS 17/24 dc^2-free codes

D=4
10111
11101
D=5
100111
101101
111001

D=6
1000111
1001011
1001101
1010011
1011001
1011111
1100101
1101001

1101111
1110001
1111011
1111101

D=7
10000111
10001011
10001101
10010011
10010101
10011011
10100011
10100101
10101001
10101111
10110001
10110111
10111011
11000101
11001001
11010001
11010111
11011101
11100001
11100111
11101011
11101101
11110101
11111001

GS 19/26 dc^2-free codes

D=5
100111
101011
110101
111001

D=6
1000111
1001011
1001101
1010011
1011001
1011111
1100101
1101001
1101111
1110001
1111011
1111101

D=7
10000111
10001101
10010011
10010101
10100011
10101001
10101111
10110001
10110111

Column 1:

```
10111011
11000101
11001001
11010111
11011101
11100001
11101011
11101101
11110101
```

GS 21/28 dc²-free codes

D=6
```
1000111
1001101
1010011
1011001
1011111
1100101
1101111
1110001
1111011
1111101
```

D=7
```
10000111
10010011
10010101
10011001
10101001
10110111
11001001
11010111
11100001
11101011
11101101
```

GS 23/30 dc²-free codes

D=6
```
1010011
1100101
```

GS 24/32 dc²-free codes

D=6
```
1000111
1001011
1001101
1010011
1011001
1011111
1100101
1101001
1101111
1110001
1111011
1111101
```

D=7
```
10000111
```

Column 2:

```
10001101
10010011
10010101
10011001
10011111
10100011
10101001
10101111
10110001
10110111
10111011
11000101
11001001
11010111
11011101
11100001
11101011
11101101
11110101
11111001
```

D=8
```
100000111
100001011
100001101
100010011
100011001
100011111
100100011
100100101
100101001
100101111
100110001
100110111
100111011
100111101
101000011
101001001
101001111
101010111
101011011
101011101
101100001
101100111
101101011
101110101
101111001
101111111
110000101
110001001
110001111
110010001
110010111
110011011
110100001
110100111
110101101
110110011
110110101
110111001
110111111
111000001
```

Column 3:

```
111001011
111001101
111010011
111010101
111011001
111011111
111100011
111100101
111101001
111101111
111110001
111110111
111111011
111111101
```

GS 2/8 dc³-free codes

D=3
```
1001
```

D=4
```
10111
```

D=6
```
1101111
1111011
```

GS 4/12 dc³-free codes

D=5
```
110101
111001
```

D=6
```
1001011
1001101
1011111
1101111
```

D=7
```
10000111
10001101
10010101
10111011
11001111
11010001
11010111
11100001
11101101
11111001
```

D=8
```
100010011
100100011
101011011
101011101
101100111
101110101
101111111
110001001
110010001
110110101
111001101
```

111111101

GS 6/16 dc³-free codes

(D ≤ 8)

D=6
1000111
1001101
1100011
1110001

D=7
10000111
10010011
10010101
10011111
10101001
10101111
10110001
10110111
11000101
11001111
11010001
11010111
11100001
11100111
11101011
11101101
11110011
11110101

D=8
100000111
100010011
100010101
100011001
100100101
100101001
100101111
100111011
101001001
101010001
101011101
101101011
101110011
101111001
101111111
110000101
110001001
110001111
110010001
110011011
110011101
110100001
110100111
110101101
110110011
110110101
110111001
111001011
111010101

111011001
111011111
111100011
111100101
111101001
111111011

GS 9/20 dc³-free codes

(D ≤ 9)

D=7
11000101
11010111
11100001
11101011
11111001

D=8
100001011
100001101
100010011
100010101
100011111
100101111
100111011
100111101
101001111
101110011
110001111
110011101
110110101
110111001
111100011
111100101
111110111

D=9
1000000111
1000001101
1000011111
1000010101
1000101111
1000110111
1000111101
1001011101
1010100001
1011001101
1011011001
1011101001
1011110001
1100001001
1100011011
1100011101
1100110101
1101001011
1101001101
1101101111
1101110001
1110000001
1110100011
1110101111
1110110001

1110111011
1111010001

GS 12/24 dc³-free codes

(D ≤ 9)

D=8
100010101
101100111
101110101
110000101
111010011
111011111

D=9
1000000111
1000001011
1000101001
1000111011
1000111101
1001000011
1001001111
1001010001
1001010111
1001110101
1001111001
1010100001
1011001101
1011111011
1100001001
1100100111
1100101011
1101001011
1101010011
1101100101
1101101001
1101111101
1110000001
1110001101
1110010011
1110100101
1110101111
1110110111
1111100001
1111100111
1111101011
1111101101
1111110101

GS 15/28 dc³-free codes

(D ≤ 9)

D=8
100010011

D=9
1000000111
1000001101
1000010101
1000011111
1000101001
1001000011

1001100001
1001100111
1001101011
1001110101
1001111001
1010010111
1010100001
1010100111
1010111001
1011000111
1011010101
1011100011
1011100101
1011101001
1011111011
1100000101
1100001001
1100011101
1100100001
1100100111
1100101011
1101001011
1101010011
1101011111
1101100101
1101101001
1101101111
1110000001
1110001011
1110011001
1110011111
1110100101
1110110111
1111000101
1111010001
1111010111
1111011101
1111101011
1111101101
1111110101

GS 18/32 dc^3-free codes

($D \leq 9$)

D=9
1000000111
1000001011
1000001101
1000010011
1000011111
1000100011
1000100101
1000101001
1000101111
1000110001
1000110111
1000111101
1001000011
1001001111
1001010001
1001011011
1001100001

1001100111
1001110101
1001111111
1010001111
1010010001
1010010111
1010011101
1010100111
1010101011
1010101101
1010110011
1010110101
1011000001
1011000111
1011010011
1011010101
1011011001
1011011111
1011100011
1011100101
1011101001
1011101111
1011110111
1011111011
1100000101
1100001001
1100010111
1100011011
1100100001
1100101001
1100101101
1101000001
1101000111
1101010011
1101010101
1101011001
1101011111
1101100011
1101101001
1101101111
1101110001
1101110111
1101111101
1110001101
1110010011
1110011001
1110011111
1110100011
1110101111
1110111011
1110111101
1111000011
1111000101
1111001001
1111010001
1111010111
1111011011
1111011101
1111100001
1111100111
1111101011
1111101101

1111110101
1111111001

GS 2/16 dc^4-free codes

($D \leq 6$)

D=3
1001

D=4
10111
11011
D=6
1000001
1011111
1100011
1101111
1110111
1111011
1111101

GS 4/20 dc^4-free codes

($D \leq 12$)

D=9
1101100101

D=10
10111100001
11000001001

D=11
100011101111
100101000111
101001001011
110000111001
111000101001

D=12
1000010000011
1000100110101
1011001001001
1011110101111
1100001101111
1100010000111
1100011101101
1101010111111
1101011011111
1101101000001
1110000101001

GS 6/24 dc^4-free codes

($D \leq 9$)

D=7
10011111
10110001
10110111
10111011
11001001
11011011
11101011

D=8
100000111
100100101
101001001
101111111
110000011
110010111
110110101
111001011
111010101
111110111
D=9
1000100011
1000100101
1000101111
1001000101
1001010001
1001011011
1001011101
1001111001
1010000101
1010001001
1010001111
1010010111
1010011011
1010011101
1010111001
1010111111
1011000001
1011000111
1011010011
1011010101
1011100011
1011101111
1011110001
1011110111
1011111011
1100000101
1100001001
1100010111
1100011101
1100110101
1100111001
1101000001
1101001011
1101011111
1101100011
1101101001
1101101111
1101111011
1101111101
1110000001
1110001011
1110001101
1110010101
1110011001
1110100101
1110110111
1111010001
1111010111
1111100001
1111101101

GS 8/28 dc^4-free codes

($D \leq 9$)

D=8
100101111

D=9
1000010011
1000011001
1000101001
1000101111
1001000101
1001010001
1001011101
1011101111
1011110001
1100010001
1100010111
1100101011
1101011111
1101110001
1110011001
1110111101

GS 11/32 dc^4-free codes

($D \leq 10$)

D=10
10100111111
10111001111
10111101101
11000100111
11000111001
11100101111

161